Department of Communications and Networking

Resource allocation in wireless access network

A queueing theoretic approach

Prajwal Osti

Aalto University

DOCTORAL DISSERTATIONS

Aalto University publication series
DOCTORAL DISSERTATIONS 137/2016

Resource allocation in wireless access network

A queueing theoretic approach

Prajwal Osti

A doctoral dissertation completed for the degree of Doctor of Science (Technology) to be defended, with the permission of the Aalto University School of Electrical Engineering, at a public examination held at the lecture hall T2 of the school on 11 August 2016 at 12.

Aalto University
School of Electrical Engineering
Department of Communications and Networking

Supervising professor
Professor Riku Jäntti

Thesis advisors
Docent Samuli Aalto, Aalto University
Docent Pasi Lassila, Aalto University

Preliminary examiners
Professor Gustavo de Veciana, University of Texas at Austin, USA
Professor Dr Rudesindo Núñez-Queija, University of Amsterdam, the Netherlands

Opponent
Doctor Konstantin Avrachenkov, INRIA Sophia Antipolis, France

Aalto University publication series
DOCTORAL DISSERTATIONS 137/2016

© Prajwal Osti

ISBN 978-952-60-6915-9 (printed)
ISBN 978-952-60-6914-2 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)
http://urn.fi/URN:ISBN:978-952-60-6914-2

Unigrafia Oy
Helsinki 2016

Finland

441 697
Printed matter

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Prajwal Osti

Name of the doctoral dissertation
Resource allocation in wireless access network: A queueing theoretic approach

Publisher School of Electrical Engineering

Unit Department of Communications and Networking

Series Aalto University publication series DOCTORAL DISSERTATIONS 137/2016

Field of research Communications Engineering

Manuscript submitted 10 February 2016 **Date of the defence** 11 August 2016

Permission to publish granted (date) 9 May 2016 **Language** English

☐ **Monograph** ☒ **Article dissertation** ☐ **Essay dissertation**

Abstract

To meet its performance targets, the future 5G networks need to greatly optimize the Radio Access Networks (RANs), which connect the end users to the core network. In this thesis, we develop mathematical models to study three aspects of the operation of the RAN in modern wireless systems. The models are analyzed using the techniques borrowed mainly from queueing theory and stochastic control. Also, simulations are extensively used to gain further insights.

First, we provide a detailed Markov model of the random access process in LTE. From this, we observe that the bottleneck in the signaling channel causes congestion in the access when a large number of M2M devices attempt to enter the network. Then, in the context of the so-called Heterogeneous networks (HetNets), we suggest dynamic load balancing schemes that alleviate this congestion and reduce the overall access delay.

We then use flow-level models for elastic data traffic to study the problem of coordinating the activities of the neighboring base stations. We seek to minimize the flow-level delay when there are various classes of users. We classify the users based on their locations, or, in dynamic TDD systems, on the direction of service the network is providing to them. Using interacting queues and different operating policies of running such queues, we study the amount of gain the dynamic policies can provide over the static probabilistic policies. Our results show that simple dynamic policies can provide very good performance in the cases considered.

Finally, we consider the problem of opportunistically scheduling the flows of users with time-varying channels taking into account the size of data they need to transfer. Using flow-level models in a system with homogeneous channels, we provide the optimal scheduling policy when there are no new job arrivals. We also suggest the method to implement such a policy in a time-slotted system.

With heterogeneous channels, the problem is intractable for the flow-level techniques. Therefore, we utilize the framework of the restless-multi-armed-bandit (RMAB) problems employing the so-called Whittle index approach. The Whittle index approach, by relaxing the scheduling constraints, makes the problem separable, and thereby provides an exact solution to the modified problem. Our simulations suggest that when this solution is applied as a heuristic to the original problem, it gives good performance, even with dynamic job arrivals.

Keywords Resource allocation, Wireless network, Queueing theory, Scheduling

ISBN (printed) 978-952-60-6915-9 **ISBN (pdf)** 978-952-60-6914-2

ISSN-L 1799-4934 **ISSN (printed)** 1799-4934 **ISSN (pdf)** 1799-4942

Location of publisher Helsinki **Location of printing** Helsinki **Year** 2016

Pages 277 **urn** http://urn.fi/URN:ISBN:978-952-60-6914-2

Preface

This thesis is the outcome of my work done at the Department of Communications and Networking, Aalto University School of Electrical Engineering between 2010–2015. The research works have been carried out as parts of AWA, EXAM, HEWINETS, HII FNS, IoT, ITTECH5G, M2MRISE and TOP-Energy projects funded by various organizations such as the Academy of Finland, Cassidian Systems, EIT Digital, EIT ICT Labs, Ericsson, Nokia-Siemens Networks, TEKES, etc.

At this point, I would like to express my sincere gratitude to my instructors Dr. Samuli Aalto—my original supervisor under whose tutelage I started my doctoral work—and Dr. Pasi Lassila for their constant and generous support and encouragement during the whole process. I would also like to thank Prof. Riku Jäntti for agreeing to be my supervisor for the dissertation. A special word of appreciation goes out to Dr. Tuomas Tirronen, Ms. Anna Larmo and Dr. Aleksi Penttinen, who, besides my advisors, have co-authored the papers compiled in this thesis. I would also like to thank the anonymous reviewers who have helped me to improve the content of these papers.

In addition, I appreciate the kind gestures of Prof. Gustavo de Veciana and Prof. Rudesindo Núñez-Queija for agreeing to be my preliminary examiners and pointing out the possible areas of improvement in the thesis after having read the initial draft. Moreover, I would like to express my sincere gratitude to Dr. Konstantin Avrachenkov for willing to be my opponent.

I am also very humbled by the great support I have received from my colleagues at the department over the years. I thank Dr. Pekka Lampio for the wonderful varieties of teas he offered during the coffee breaks and his discussions about various random topics. My special thanks go out to the secretaries Ms. Sanna Patana, Ms. Sari Kivelö and Ms. Heli Liukko for

their help in administrative and official matters, and to Mr. Victor Nässi for providing every kind of support, always with a broad smile on his face. Furthermore, I thank Ms. Marja Leppäharju and Ms. Anne-Mari Karro from the Student Service for making the dissertation process so smooth.

I would like to express my deepest gratitude for the warm friendship of many people here in Finland, who have provided the much needed distraction from the work at appropriate times through various forms of social gatherings. I am also humbled by the generosity of the Finnish people for their support when carrying out my studies, and for providing me an opportunity to broaden my horizons by introducing me to a such a different way of life than I was used to. Finally, I would like to thank my parents and sister—Mamu, Baba and Lora—for always being there for me, even if they live thousands of kilometers away.

Espoo, July 8, 2016,

Prajwal Osti

Contents

List of Publications

This thesis consists of an overview and of the following publications which are referred to in the text by their Roman numerals.

I Prajwal Osti, Pasi Lassila, Samuli Aalto, Anna Larmo, Tuomas Tirronen. Analysis of PDCCH performance for M2M traffic in LTE. *IEEE Transactions on Vehicular Technology*, Vol. 63, No. 9, pages 4357–4371, November 2014.

II Prajwal Osti, Samuli Aalto, Pasi Lassila. Load balancing for M2M random access in LTE HetNets. In *Proceedings of IEEE 22nd International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2014)*, pages 132–141, Paris, France, September 2014.

III Prajwal Osti, Samuli Aalto, Pasi Lassila. Minimizing access delay for M2M traffic in multi-RAT HetNets. In *Proceedings of the 18th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM'15)*, pages 161–170, Cancun, Mexico, November 2015.

IV Prajwal Osti, Samuli Aalto, Pasi Lassila. Optimal intercell coordination for multiple user classes with elastic traffic. In *Proceedings of the 8th EURO-NGI Conference on Next Generation Internet (NGI 2012)*, pages 25–32, Karlskrona, Sweden, June 2012.

V Prajwal Osti, Samuli Aalto, Pasi Lassila. Flow-level modeling and opti-

mization of intercell coordination with dynamic TDD. In *Proceedings of the 10th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet'14)*, pages 109–118, Montréal, Canada, September 2014.

VI Prajwal Osti, Samuli Aalto, Pasi Lassila. Performance and optimization of dynamic-TDD intercell coordination with elastic traffic. Accepted for publication in *Telecommunication Systems*, 2016.

VII Samuli Aalto, Aleksi Penttinen, Pasi Lassila, Prajwal Osti. On the optimal trade-off between SRPT and opportunistic scheduling. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '11)*, pages 185–195, San Jose, USA, June 2011.

VIII Samuli Aalto, Aleksi Penttinen, Pasi Lassila, Prajwal Osti. Optimal size-based opportunistic scheduler for wireless systems. *Queueing Systems*, Vol. 72, No. 1, pages 5–30, March 2012.

IX Samuli Aalto, Pasi Lassila, Prajwal Osti. Whittle index approach to size-aware scheduling with time-varying channels. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '15)*, pages 57–69, Portland, USA, June 2015.

Author's Contribution

Publication I: "Analysis of PDCCH performance for M2M traffic in LTE"

The publication is a joint work between the authors. The idea originally came from Anna Larmo at Ericsson. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author.

Publication II: "Load balancing for M2M random access in LTE HetNets"

The publication is a joint work between the authors. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author. The author drafted the paper.

Publication III: "Minimizing access delay for M2M traffic in multi-RAT HetNets"

The publication is a joint work between the authors. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author. The author drafted the paper.

Publication IV: "Optimal intercell coordination for multiple user classes with elastic traffic"

The publication is a joint work between the authors. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author. The author drafted the paper.

Publication V: "Flow-level modeling and optimization of intercell coordination with dynamic TDD"

The publication is a joint work between the authors. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author. The author drafted the paper.

Publication VI: "Performance and optimization of dynamic-TDD intercell coordination with elastic traffic"

The publication is an extension of V. The model was developed by the author together with Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author.

Publication VII: "On the optimal trade-off between SRPT and opportunistic scheduling"

The publication is a joint work between the authors. A version of Theorem 1 was proved and the results of Section 6.1 were obtained by the author in his master's thesis with the help of Dr. Samuli Aalto.

Publication VIII: "Optimal size-based opportunistic scheduler for wireless systems"

The publication is an extension of VII. A version of Theorem 1 was proved by the author in his master's thesis with the help from Dr. Samuli Aalto. The flow-level simulations related to OPS* policy in Scenario A and B were carried out by the author.

Publication IX: "Whittle index approach to size-aware scheduling with time-varying channels"

The publication is a joint work between the authors. The model was developed by Dr. Samuli Aalto and Dr. Pasi Lassila. The simulations and computations were carried out by the author and Section 8 was drafted by the author

List of abbreviations

3GPP	Third Generation Partnership Project
4G	Fourth generation
5G	Fifth generation
BF	Balanced fair
BSO	Both Stations On
CCE	Control Channel Element
CN	Core network
CQI	Channel quality index
CRAN	Cloud Radio Access Network
CSMA	Carrier sense multiple access
D2D	Device-to-device
DL	Downlink
eNB	eNodeB
EVDO	Evolution-Data Optimized
FP	Flow-level problem
FPI	First policy iteration
H2H	Human-to-human
HD	Downlink priority policy
HDR	High Data Rate
HetNet	Heterogeneous network
HU	Uplink priority policy

IoT	Internet of Things
IP	Internet Protocol
JSQ	Join shortest queue
LTE	Long Term Evolution
M2M	Machine-to-machine
MAC	Media access control
MCSA	Multichannel Slotted Aloha
MDP	Markov Decision Processes
MW	Max-Weight
OFDM	Orthogonal frequency division multiplexing
OP	Original problem
PDCCH	Physical Downlink Control Channel
PDSCH	Physical Downlink Shared Channel
PF	Proportionally fair
PI	Policy iteration (Chapters 1,2,3)
PI	Potential Improvement (Chapter 4)
PRACH	Physical Random Access Channel
PS	Processor-sharing
PUSCH	Physical Uplink Shared Channel
RA	Random access
RACH	Random Access Channel
RAN	Radio Access Network
RAP	Radio Access Point
RAT	Radio access technology
RMAB	Restless multi-armed bandit
RP	Robust Policy (Chapter 3)
RP	Relaxed problem (Chapter 4)
SRPT	Shortest remaining processing time
SRPT-FM	Shortest remaining processing time for the fastest machine

SW	Size-aware Whittle index
TCP	Transmission Control Protocol
TDD	Time division duplexing
UE	User equipment
UL	Uplink
WLAN	Wireless LAN

List of symbols

Common operators

$\mathbf{E}[X]$	Expected value of random variable X
$\mathbf{P}\{E\}$	Probability of event E
$\mathbf{Var}[X]$	Variance of random variable X
$\boldsymbol{x}$	A vector
$\lvert\boldsymbol{x}\rvert$	Sum of the components of $\boldsymbol{x}$

Chapter 2

Latin symbols

a	Aggregate arrival rate of the random access requests
a_2^*	Aggregate arrival rate maximizing Message 2 throughput
a_4^*	Aggregate arrival rate maximizing Message 4 throughput
A_n	Number of new random access requests in timeslot n
b	RACH periodicity
c	Maximum number of UL grants per subframe in Message 2
D_n	Number of non-colliding preambles in timeslot n
K	Number of preambles
m	Maximum number of Message 4's per subframe with Message 2's (Section 2.3)
m	Number of femtocells (Section 2.4)
M	Maximum number of Message 4's per subframe without Message 2's
N	PDCCH resource size in CCEs
N^{Msg2}	Number of CCEs used for a Message 2

N^{Msg4}	Number of CCEs used for a Message 4
p_i	Fraction of users staying in cell i
$r(n)$	Retransmission probability with backlog n
X_n	Backlogged Message 4's in time slot n
$X_i(t)$	Backlog of cell i at time t (Section 2.4)
$\bar{X}_i(\lambda)$	Average backlog in cell i with arrival rate λ
$Y_n^{(1)}$	Number of successful Message 1's
$Y_n^{(2)}$	Number of non-colliding UL grants included in Message 2's
$Y_n^{(3)}$	Number of successful Message 3's
$Y_n^{(4)}$	Number of transmitted Message 4's
$\tilde{Y}_n^{(1)}$	Number of preambles chosen
$\tilde{Y}_n^{(2)}$	Number of UL grants included in Message 2's

Greek symbols

$\theta^{(1)}(a)$	Throughput of successful Message 1's per subframe
$\theta^{(2)}(a)$	Throughput of successful UL grants per subframe
$\tilde{\theta}^{(2)}(a)$	Throughput of all UL grants per subframe
θ^*	Maximum throughput of Message 4's
λ	Arrival rate of fresh random access requests
λ_i	Arrival rate of fresh random access requests in cell i
μ	Average service rate of the femtocells
μ_0	Average service rate of the macrocell
$\sigma^{(4)}(a)$	Average left-over capacity of Message 4's per subframe
τ	Length of the duration of the timeslot

Chapter 3

Latin symbols

$a^\pi(\boldsymbol{x})$	Action chosen under policy π in state $\boldsymbol{x}$
B_i^l	Random job size of type-l user associated with Station i
B_i^δ	Random job size of class (δ, i)
$c_i^m(r)$	Average transmission rate of a user at distance r
i	Index of cell/station
k_i^m	Ratio of the service rates of far to near users
l	Location of user in cell
m	Operating mode
m_i	Direction of service of Station i in mode m (ith letter of m)

p^m	Probability of choosing mode m
R_i	Random distance of a user
R_i^l	Random distance of an l-type user
$X_i^l(t)$	Number of flows of l-type at time t
$X_i^\delta(t)$	Number of flows in class (δ, i) at time t

Greek symbols

δ	Direction of service
λ_i^l	Arrival rate of flows located at l
λ_i^δ	Arrival rate of class (δ, i)
μ_1	Service rate of a base station when both are serving
μ_0	Service rate of a base station when the other is turned off
μ_i^m	Average service rate when operating mode m
$\mu_i^{m,l}$	Average service rate of an l-type user
ϕ_i	Service rate with the static policy
ϕ_i^*	Service rate with the optimal static policy
ϕ_i^δ	Service rate of class (δ, i)
$\phi_i^{*\delta}$	Optimal static service rate of class (δ, i)
$\phi_i^\delta(\boldsymbol{x})$	Balance fair allocation in state $\boldsymbol{x}$
$\Phi(\boldsymbol{x})$	Balance function for state $\boldsymbol{x}$

Chapter 4

Latin symbols

$A_i^\pi(t)$	Variable indicating if a user is scheduled at timeslot t
c_i	Holding cost rate of user i
$c_{n,i}$	Long term service rate of user i when there are n users
$\boldsymbol{c}^\phi$	Operating point under policy ϕ
$\mathcal{C}_n$	Opportunistic capacity region with n users
f^π	Average of total cost under policy π
f_β^π	Average of discounted cost under policy π
i	Index of user
J	Total number of phases that in the approximation of x_i
K	Number of users initially in the system
r_i	Channel state of user
$r_{i,1}$	Lower channel state
$r_{i,2}$	Higher channel state

$\bar{r}_i$	Average channel state
$R_i(t)$	Channel state in timeslot t
t	Index of timeslot
$\boldsymbol{w}^{\phi}$	Weight-based scheduler under policy ϕ
x_i	Initial size of job
X_i	Random variable approximating x_i
y_i	Remaining job size
$Y_i^{\pi}(t)$	Remaining job size at the beginning of timeslot t
$Z_i^{\pi}(t)$	Number of phases of job at the beginning of timeslot t

Greek symbols

β	Discount factor
γ_n	Opportunistic gain with n users
$\nu^*(j,r)$	Whittle index of a job in phase j and channel state r
$\nu_i^{\pi}(y_i,r_i)$	Primary index
$\tilde{\nu}_i^{\pi}(y_i,r_i)$	Secondary index
π	Scheduling policy
π^*	Optimal scheduling policy
Π_n	Set of all scheduling policies with n users
τ	Duration of timeslot
ϕ	Operating policy
ϕ^*	Optimal operating policy

1. Introduction

Increasing the data rate has been the primary focus in the development of each new generation of cellular wireless networks. As high-speed wireless Internet has become more ubiquitous with the deployment of the Long Term Evolution (LTE), the fourth generation (4G) wireless network, new communication applications like the Internet of Things (IoT) and Machine-to-machine (M2M) communications are emerging. To support these new services, the network needs to be scalable and should ensure features such as low latency, the ability to support a massive number of connected users, low power consumption, overall energy efficiency, ultra reliability, seamless mobility, etc. These features should be supported in addition to the capability for high data rates, whose demand is predicted to grow exponentially in the foreseeable future [2, 39].

To cope with the new service needs, advanced features are being incorporated into the standards with each new release of LTE. However, supporting all the services predicted to exist may not be possible just by adding a few enhancements to the existing technology. Therefore, a newer 5th generation wireless network (5G) is envisioned to be standardized by the year 2020 that is able to provide these services [46, 62]. This upgrade of technology is primarily aimed at the Radio Access Network (RAN), which is directly connected to the user equipments (UEs) [66]. As the Core Network (CN) is already packet switched and uses the Internet Protocol (IP) in most of the cases, moderate changes may be sufficient there to cope with the additional demand.

Methods such as extending wireless communication to the currently unused frequencies [71] and using cooperative communication [96] techniques in the unlicensed frequency bands are being proposed for 5G radio access. The proposed newer frequencies are higher than the ones currently in use (can extend up to several tens of Gigahertz), have wider

bandwidth, and, together with the existing frequencies, support higher data rates. Additionally, developments in cognitive radio also allow spectrum sensing in the unlicensed bands. This helps to efficiently utilize these bands for providing additional data capacity, while the signaling is carried out in the licensed band [39].

Moreover, the use of lean design and zero overhead communications [39] to exchange M2M messages, which are typically very small, are also being considered to be incorporated in 5G. This can help to achieve the low-latency requirements also necessary for certain applications like emergency services. Short range Device-to-device (D2D) [84] communications also aids in achieving the low latency target. In D2D, the network may assist the devices to establish a link using which they can communicate with each other directly without using the network resources.

One of the most promising techniques to address the increased need of data capacity is the deployment of Heterogeneous networks (HetNet) [41], in which there are multiple low-power *small* cells (also called microcells, picocells, femtocells or phantom cells in different contexts [42, 64]), inside the coverage areas of high-power macrocells. Consequently, the small cells are able to serve only the local traffic hotspots, while the macro serves the whole area. Furthermore, these smaller cells may or may not be based on the same technology as the larger cell. Therefore, seamless integration of these two is one of the main challenges in future network design.

Similarly, flexible duplex transmission methods such as dynamic time division duplexing (TDD) can also be used to meet service demands [64]. In dynamic TDD, the number of uplink and downlink timeslots can be dynamically changed depending on uplink and downlink traffic conditions [85]. Consequently, the transmission powers of the stations and the UEs should be tightly controlled, especially in small cells, where the terminals and the base stations are transmitting at almost the same power levels.

Technologies such as Cloud-RAN [32] or Centralized-RAN (CRAN) [78] have been proposed as a way of operating these 5G base stations. They help in mitigating possible interference, where a CRAN controls many Radio Access Points (RAPs). With the development of Network Service Virtualization and Software Defined Network technologies, the CRANs can be constructed from non-specialized commodity hardware [91]. Moreover, these technologies also aid the development of RAN as a service (RANaaS) [78] concept, whereby the network functions implemented in CRAN and the RAPs may be easily transferred to one another depending on deploy-

ment scenario.

Such enhancements help to meet some of the challenges posed in the visions of 5G. On the other hand, when massive number of machines try to communicate, the current random access technique quickly leads to congestion, partly due to passing many messages prior to sending the actual data [3, 54]. The actual messages they send may even be smaller than these overhead messages, and zero overhead communication certainly helps. With the use of CRAN, a central control over the whole network can be maintained and users can be instructed to join the appropriate cell in HetNet depending on the congestion conditions thereby reducing the access delay. Moreover, CRAN also allows for proper intercell coordination, where the transmission powers of the base stations are strategically controlled. This aids in mitigating the intercell interference and thereby enhances the data rate. In addition, using good scheduling policies even in a single cell environment can increase the data rate for the users [61]. If the scheduling policies can exploit more information such as the channel condition, the size of data, the type of data, etc., to select the user that can be best served then it is obvious they may provide better performance. At the same time maintaining fairness, i.e., not completely ignoring the users with unfavorable scheduling conditions, like those at the cell-edge, is also an important issue.

In this thesis, we focus on the features that are potentially available in the next generation network to optimize the traffic performance at an abstract level using various mathematical tools. The traffic consists of users that wish to utilize the access network's resources to fulfill their service demands. The service demand can be just the entry to the network, or it may involve the transmission of some data in either uplink or downlink directions. We use techniques from queueing theory and stochastic control to construct mathematical models of different aspects of modern wireless communications systems. These generic models are then applied to either the standardized 4G or the future 5G networks, to learn about some specific aspects of the network resource allocation. The models are useful in analyzing the possible shortcomings of these allocations. Moreover, by using different optimization techniques on these models, we are able to suggest potential improvements.

1.1 Research problems

We mostly focus on the operation and optimization of the access network in modern or possibly future wireless systems from the traffic point of view. More precisely, we delve into three traffic-related aspects of the operation of these networks.

Random access in LTE
First, we study the congestion problem at the LTE random access, which is faced by all the users wishing to enter the LTE network. However, the problem becomes more prominent only when a large number of machines attempt to enter the network for M2M communications [54]. Traditionally, research has mostly focused on increasing data rate, assuming that there are enough signaling resources that make the data transfer possible.

In LTE, a number of messages are exchanged between the UEs and the base station using the signaling channel, PDCCH, during the random access stage [40]. The system may have enough capacity to handle the data requests from these machines as they are very small and intermittent. However, due to the limitations of PDCCH, there is not enough capacity to allow all of them into the network, making any subsequent data transfer impossible [3, 54]. In this thesis, we attempt to find the limitations of the random access process in LTE due to the limited capacity of the signaling channel, taking into account the various messages that are exchanged between the UE and the base station. We seek to find the maximum rate of the random access requests that can be sustained under the PDCCH limitations. When the resource constraints make the random access requests unsuccessful, we try to identify the exact causes of such failures and calculate their respective probabilities as well. The findings can also help us to dimension the network for the number of users that the system can reliably sustain.

HetNets have been proposed to enhance the network capacity [41]. They also aid in relieving the random access congestion. In HetNets, femtocells serve the local traffic hotspots and the rest of the users are served by the macrocell alone. Since the macrocell behaves as a kind of overflow channel, which every user of the femtocells can also access, we attempt to find the best decision a user coming to the femtocell can make, i.e., should it attach itself to the femto or the macro so that the access delay is minimized? The answer depends on the amount of information we have. In

general, better performing policies can be devised with more information. Thus, we can construct various heuristic policies that appear intuitively good and systematic policies suggested by the theory of Markov Decision Processes (MDP). These policies need different amounts of information about the network conditions. The information includes the arrival rates of traffic in different cells, the service rate of traffic there, and even the instantaneous congestion condition given by the number of users queued to obtain service from the cells. Depending on the technologies used at the macro and the femtos, we can experiment with various policies and attempt to find out which of them performs the best for minimizing the access delay.

Intercell coordination

After studying the issues related with the access, we shift attention to the data transfer stage. To study the transmission of elastic traffic, such as the downloads of files from servers, flow-level models are used [21, 25, 28, 29, 43, 94]. Such models characterize the dynamic behavior of the system at the time-scale of flow arrivals and departures. In this case, the processor-sharing (PS) queueing models provide a very useful abstraction of the resource allocation among the flows in a wireless systems.

These models are used to study the intercell coordination problem in a multicell environment, i.e., how to gain from jointly controlling the activities of the base stations? We specifically consider two neighboring cells that interfere with each other. This means that each cell operates at a higher rate if the other is turned off. This situation with elastic traffic is analyzed using a set of interacting PS queues.

First we consider users distinguished by their location in the cell, i.e., there are near and far users, and the near ones are served at a rate higher than the ones far-off. We seek to find an optimal dynamic coordination policy, which takes into account the location of users, to operate the cells so that the average flow-level delay is minimized. Such policies can be determined using the MDP approach.

In addition, we consider two base stations using dynamic TDD, where the users are differentiated by their service directions (uplink and downlink) so that there are four different operating modes. We seek different dynamic coordination strategies between the cells, so that the average flow-level delay is minimized, using MDP and other approaches.

Opportunistic scheduling

Finally we consider the problem of combining channel and flow-size information for scheduling in a single cell. The system is again analyzed at the flow-level for elastic traffic. When multiple flows are present in a cell, the base station's resources can be scheduled by taking the stochastically varying channel conditions of the flows into account. Thus, when a large number of users with randomly varying channels are present, it is very likely that there is at least one user with good channel conditions, whose flow can then be opportunistically scheduled for service. Clearly, this kind of multiuser diversity increases the throughput of the system and leads to opportunistic scheduling gain. On the other hand, scheduling flows based on the information about their sizes is also useful in reducing the flow-level delay. Most notably, a classic result [80] from the queueing theory suggests that, in a system with constant service rate, always serving the flow with the smallest remaining size is the optimal way to schedule.

Thus, we have a clear trade-off between size-based scheduling and opportunistic scheduling in a wireless system: Scheduling the flow with the smallest remaining size at all times deprives the system of the possible opportunistic gain because the smallest flow may not always have the best channel conditions. On the other hand, a purely opportunistic scheduler, that takes into account the channel states of all the users and serves the flow with the best channel conditions, forfeits the possible gain that can be extracted from the size information. Now, the research problem is to find the best possible ways to combine the two scheduling principles by making the appropriate trade-offs.

A flow-level description of the problem naturally leads to the answers about optimal scheduling of the flows at the time scale of the arrival and the departure of the flows. However, the systems actually work in a much shorter time scale called timeslot. Thus, we also need to translate the optimal flow-level scheduling rules to the timeslot level, at which the schedulers actually work.

1.2 Methodology

We study these problems related to the application of queueing theory and stochastic control to optimize the operation of RAN in a modern wireless system. We mainly apply the classic techniques and results from queueing and scheduling theories to define, as well as to solve these problems.

We make use of flow-level models [28] to study the intercell coordination problem in neighboring cells and the opportunistic scheduling of users in single cell. The request of service to a network arrives in the form of flows from users, and we are mostly interested only in the end-to-end delay time of flows. Additionally, we also assume that such flows consist of elastic traffic, i.e., protocols such as TCP are governing the transfer of these flows in the network, and therefore, they permit considerable variability in service rates. The assumption of time-scale separation [28] plays a central role in such flow-level models. In timeslotted wireless systems, we have a clear distinction between two different time scales—the interval between the successive arrivals and departures of flows, where the flow-level dynamics occur, and the duration a flow is scheduled at a time, i.e. the timeslot during which the fluctuation of the channel is relatively low. We assume that these two time scales are separated by several orders of magnitude. This allows us to consider the average service rate of the flows, instead of instantaneous service rates, and therefore paves the way for using PS-type queueing models to study the behavior of the system.

In many of the cases, we describe our system using Markov processes or chains with multi-dimensional state-spaces, which are then solved either analytically, or numerically or through simulations to study the behavior of the modeled system. We also apply the dynamic programming techniques in the context of Markov Decision Processes (MDP) [70], to optimally control the system. In particular, we apply the Policy Iteration (PI) algorithm from the MDP theory. In multidimensional Markov processes, we observe that the first step of PI (called the first policy iteration or FPI) is relatively easy to apply, if the the multi-dimensional chain is decomposed into a group of independent one-dimensional chains. In many cases, it has been observed that FPI is able to provide good results. However, after the first iteration, the curse of dimensionality can no longer be avoided. Thus, to fully carry out the PI algorithm, we make use of numerical methods for truncated Markov processes, and obtain an approximately optimal policy in certain cases. Although these techniques are accurate only up to a certain extent, they provide good insights about the actual optimal policies. We also consider various priority policies, like the well known Max-Weight [86], from the literature to study our scheduling problems in interacting queues.

Since we model rather complex systems using different approaches, the study is often carried out using dynamic simulations. The tool that is

used mostly throughout the thesis is Wolfram Mathematica, which has also been utilized to obtain certain analytical results.

1.3 Contributions of the thesis

This thesis summarizes the work done in three aspects of the access part of a modern wireless network:

Random access in LTE
We provide a model of the four-steps of LTE random access procedure using Markov chains. The stability analysis of the chain reveals the bottleneck of the procedure in PDCCH. We further determine the maximum throughput of the random access process, and calculate the probabilities of failure of the random access procedure due to the different causes.

For M2M communications in HetNets, we explore different load balancing strategies. We provide a Markov model of the random access step in such HetNets. The methods from MDP are used to determine the (near) optimal load balancing policies. Moreover, we also test various heuristic policies that can balance the load among the stations so that the access delay in the network is minimized. We first study these load balancing techniques when both the macrocells and the femtocells are based on the same LTE technology. We also extend the results to HetNets where the macro is LTE-based while the femtos operate with WLAN.

Intercell coordination
We use the flow-level models assuming the separation of time-scales between flow dynamics and scheduling decisions to investigate the intercell coordination problem between neighboring base stations. We analyze the problem using PS queues when there are various classes of users based on location or service direction.

In a setting with two neighboring base-stations with identical service rates and low interference, which serve their users at a rate based on their locations in the cells, we show that using both the stations as much as possible is a very good strategy. We provide evidence that this strategy performs better than a systematic optimization technique from the MDP theory.

In a setting involving two neighboring base stations, which use dynamic TDD to serve uplink and downlink users, we provide a flow-level model

with interacting queues and characterize the maximal stability region. We explore methods to control these interacting queues so that the flow-level average delay of uplink and downlink flows is minimized. We prove that certain priority based policies are stochastically optimal, under some assumptions. However, in general, we notice that such policies perform poorly and become unstable quite quickly. Therefore, we investigate other policies, which are, e.g., based on the MDP theory or known to give good performance in similar contexts.

Opportunistic scheduling

We consider the problem of scheduling users in a transient setting, where no new user arrives. We use flow-level models to derive policies that give the minimum average delay under different channel conditions and various scheduling assumptions. These results are then applied as heuristics in a dynamic setting, where new users can arrive at any time.

More specifically, we first consider a symmetric case where users have identical channels. Using the time-scale separation assumption and flow-level models, we characterize the optimal trade-off between opportunistic scheduling and size-based scheduling at the flow level. We also give a corresponding timeslot-level implementation technique for making such trade-offs. The performance of this method is tested in a system with dynamic flow arrivals, where we observe that this method, although no longer optimal, nevertheless provides good performance.

Secondly, we provide a heuristic size-aware opportunistic scheduler in a system that has users with heterogeneous channels. We first approximate the job sizes by the shifted Pascal distribution, and then formulate the scheduling problem as a restless-multiarmed-bandit (RMAB) problem. This is then solved using the *Whittle-index* approach. The resulting policy is provably optimal in a transient setting with relaxed scheduling constraints. The relaxed constraints allow us to schedule one user on *average* instead of exactly one user in one timeslot. However, through simulations, we observe that a heuristic policy based on the Whittle index approach provides very good performance even in the dynamic setting for the original problem, where exactly one user is scheduled in one timeslot.

1.4 Structure of the thesis

The thesis is organized as follows: In Chapter 2, we discuss the topic of M2M random access. In Chapter 3, we discuss the topics relevant to intercell coordination between neighboring cells. In Chapter 4, we discuss opportunistic scheduling. Finally we draw the conclusions of the thesis in Chapter 5.

2. Machine-to-machine random access

2.1 Introduction

Machine-to-machine (M2M) communications refers to the technology that allows autonomous communication between various machines which are usually low-power sensors [33]. With the introduction of the Internet of Things (IoT) [38] billions of such devices will start to use the existing network infrastructure in the near future [1], and as LTE is the standard for the fourth generation cellular wireless network, it has to be able to accommodate all the users that want to use the network, including the M2M devices.

M2M communication presents additional challenges to the design of the network. The network has been optimized to handle more and more data traffic with each successive generation. However, the M2M traffic and its service requirements are fundamentally different [83] from the traditional *human-to-human (H2H)* traffic that is historically more data-and-voice-centric than that the current network is primarily expected to handle. The M2M devices tend to be static, may access the network periodically or in bursts and have very low power available for communications. Although each one of them has a very small service requirement, when they are present in extremely large numbers, they can easily overwhelm the network, which can render the network unusable due to congestion.

The problem of congestion originates from the so-called random access stage. It is the first step that a device must go through if a wireless device desires to establish a connection with the base station. More interestingly, the random access process makes use of the signaling channel (PDCCH) in LTE and due to the large number of random access attempts, this channel may become congested. Traditionally, much research has focused on

optimizing the usage of the data channel assuming that enough signaling resources are available. However, with the advent of M2M communications, the congestion in the signaling channel also has to be taken into account for the proper operation of the network.

The basis of random access is the Slotted Aloha protocol, which has been studied very actively since its invention [76]. However, Slotted Aloha is fundamentally unstable, and it has to be stabilized to make it work [16]. If a transmission attempt fails in Slotted Aloha due to simultaneous transmissions from two or more devices, then all of them can make retransmission attempts, which further increases the traffic and worsens the network performance. Multichannel Slotted Aloha (MCSA) has been suggested as a remedy in which more than one Slotted Aloha channel work in parallel. However, the fundamental problem of instability still remains. Moreover, there are more steps following the Slotted Aloha step in LTE random access, which can further limit the network's capability to handle the M2M devices, especially when their number is very high.

To alleviate the problem of congestion, the future networks have envisioned the so-called Heterogeneous networks (HetNets), in which inside the coverage area of a high power macro station (called the macrocell) there are other transceivers that provide the coverage for local traffic hotspots called the femtocells. They can relieve the RACH congestion problem to some extent if they are operated properly and the traffic is directed to the macro or the femtocell depending on the congestion situation of the network.

In this chapter, we first show how the PDCCH of LTE gets congested when a large number of devices attempt to access the network which is the topic of Publication I. Then we study the operation of HetNets which can help the network to handle the random access channel (RACH) congestion issues by diverting an appropriate amount of the traffic from the femtos to the macro. This is the topic of Publication II and Publication III where we deal with the HetNets with similar RATs (LTE in this case) and different RATs (LTE for macro and WLAN for femto), respectively.

2.2 Related work

Aloha was first developed at the University of Hawaii in the late 1960s by Abramson [7] and the Slotted Aloha protocol was then introduced a few years later [76]. The Markov model and the method of stabilization of

Slotted Aloha is discussed in [16]. The techniques for stabilizing Slotted Aloha for random access have also been proposed in [75, 97]. We can see from the 3GPP specification [4] that Slotted Aloha is the basis of the first stage of the random access procedure in LTE. The issue with the congestion of the random access overload in LTE, in which Slotted Aloha plays a big role, is studied in [59, 54]. Moreover, the use of different backoff mechanisms and their performance analysis have been done in [81, 82], which have been proposed as a way to relieve congestion in LTE random access.

In [59], the authors have presented a nice overview of the problems associated with allowing a large number of machines to access the network. Various methods to overcome the problem have also been proposed including dynamic sharing of preambles between H2H and M2M traffic [57], dividing the traffic into different classes and giving priority of random access according to the classes (access class barring scheme) [35], increasing the frequency of the random access subframes [37], the use of capillary network approach [34], sending the small data without the need of much signaling [11], etc. Most of these studies focus just on the first step of the random access procedure—the Slotted Aloha. There are more steps in the random access process and only a few studies [54, 3] have been done to assess the effects of the latter stages of the random access procedure on the overall access throughput. These studies mainly focus on system-level simulation. To augment their approach, we propose a relatively simple model that takes into account these later steps as well in this thesis, which has been published as a part of Publication I.

When we move from single cell to multiple cells in the HetNet environment, the research has only recently started as most of the analysis tend to focus on single-cell environment. The Markov model of the backlogs needed for analysis in such cases have been studied in [53, 44, 77, 88] and more recently in [81, 45, 8] for LTE's RACH. We use, in Publication II, similar techniques to model the backlog of the LTE-operated cell that can then be used to perform various optimizations. Moreover, the use of WLANs as femtos is appearing to be a more viable option as suggested by [72, 10]. To study the load balancing with WLAN femtos, in Publication III we use a model of CSMA algorithm similar to those that have been used in [17, 22, 20].

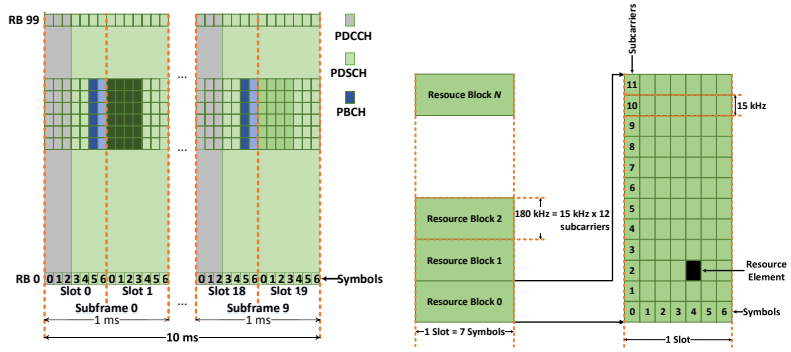


Figure 2.1. LTE resource structure

2.3 PDCCH congestion

The congestion in the Physical Downlink Control Channel (PDCCH) of LTE during the random access attempt by a large number of machines is the main topic of Publication I, where we discuss the bottleneck and the limitations of PDCCH during the random access (RA) procedure in LTE. The details of the operation of LTE are outside of the scope of the thesis and in what follows we discuss the aspects of LTE pertinent to our work. The detailed explanation of the whole LTE system can be found in [40].

In LTE, OFDM is the basic scheme of transmission for both the downlink and uplink directions. The basic OFDM subcarrier spacing is 15 kHz and transmissions are organized into *frames* of length 10 ms, each of which is further subdivided into ten 1 ms long *subframes*. A resource block consists of 12 consecutive resource subcarriers in frequency domain and one 0.5 ms time slot in the time domain. The basic time-domain unit for dynamic scheduling in LTE is one subframe (consisting of two consecutive 0.5 ms slots). The carrier can consist of any number of resource blocks between six and 110 in the frequency domain according to specifications. This is illustrated in Figure 2.1. Additionally, various logical and physical channels used for signaling and data transfer in LTE are also illustrated.

2.3.1 Research problem

Random access allows a terminal to setup a connection with the network through the eNB. It is also performed in LTE for reestablishing a radio link after failure, performing handover between cells, and various other reasons. To establish the initial access, as discussed later in Section 2.3.3,

the UE and the eNB exchange a number of messages, which consume some resource of the data and signaling channel (and ultimately consist of some LTE time-frequency resources). Since the messages used in the random access utilize the same pool of resources, when a large number of devices attempt to enter the network, the signaling channel may get congested. Naturally, the presence of a very large number of M2M devices can easily overwhelm the network because the random access process has been designed primarily to accommodate the human traffic. This congestion causes the random access to fail and reduces the access throughput.

In this section, we analyze the random access process from the limitation of the signaling channel's point of view. For the analysis of the PDCCH, we mainly use a discrete-time Markov chain model. We determine, analytically and through simulations, the probability of the occurrence of various failure events and the average throughput of the access requests as a function of the arrival rate of these requests. The analysis also reveals the cause of the bottleneck and allows us to observe the extent of deterioration of throughput caused by the increase in the rate of random access requests.

2.3.2 LTE random access procedure

For uplink data transmission, the contention based random access procedure is used in LTE. This consists of the exchange of different messages—Message 1–Message 4—between the UE and the eNB as illustrated in Figure 2.2. The process is said to be successful if these messages are successfully sent and received within some deadline specified by the standards. A successful random access process ensures that the UE can transmit data using the network's resources as described in detail in [40]. The aspects of the procedure that are relevant to our model are briefly described below.

In **Step 1**, the UE randomly chooses one of the K available Random Access Channel (RACH) preambles and sends it as a Message 1 over the Physical Random Access Channel (PRACH). A preamble is a Zardoff-Chu sequence [40], and 64 of them are potentially available for the UEs to use. The network, using the broadcast channel, informs the UEs about the location of PRACH (in the time-frequency resource) to be used for random access. A collision occurs if two or more users transmit the same preamble, and such preambles are called *collided* preambles. However, we later see that the collisions are detected only in Step 3.

In **Step 2**, the eNB detects a preamble transmission from the UE and

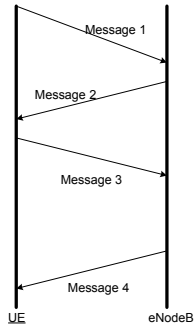


Figure 2.2. LTE RA procedure sequence

immediately responds with the Random Access Response (Message 2) in the PDCCH and uplink grants, i.e., the schedule for sending Message 3 in the Physical Downlink Shared Channel (PDSCH) for every *used* preamble (collided or uncollided).

In **Step 3**, after receiving the Message 2, the UE sends a Message 3 over the Physical Uplink Shared Channel (PUSCH). The collisions in Step 1, if any, are now realized as all the UEs that chose the same preamble in Step 1 try to use the same uplink grant to send their Message 3's. As a result, the Message 3's belonging to the UEs that chose the same preamble cannot be decoded by the eNB. The eNB does not send the subsequent Message 4 to these UEs.

Finally, in **Step 4**, after receiving Message 3's related to the uncolliding preambles the eNB replies with Message 4's using the PDCCH. A buffer is used by the eNB to queue messages that cannot be immediately sent back to the UE.

After the successful reception of Message 4, the UE proceeds to send the data using the data channel. The unsuccessful UEs attempt a retransmission after some backoff time, which typically increases with the failure of every retransmission attempt. If the retransmission is unsuccessful even after many attempts, the request is dropped.

2.3.3 Model and analysis

We proceed to model the evolution of Message 4 buffer, with a few more assumptions, taking into account all the four steps described in Section 2.3.2 using a two-dimensional Markov chain. This model can then be used to determine the maximum throughput $\theta_{\max}$ of the random access requests

that a system can sustain. We define a *timeslot*[1] as the duration of one random access opportunity that consists of b subframes. One preamble corresponds to one Slotted Aloha channel which can be accessed only at the beginning (the first subframe) of a timeslot. We assume that there are K preambles which are randomly selected by the UEs, making the first step of the random access process effectively a Multichannel Slotted Aloha (MCSA) system with K parallel channels. Every preamble which has been used by just one UE is referred to as a *successful preamble* while the ones used by at least one UE are referred to as the *chosen preambles*. Their numbers in timeslot n are denoted by $Y_n^{(1)}$ and $\tilde{Y}_n^{(1)}$, respectively and obviously $Y_n^{(1)} \leq \tilde{Y}_n^{(1)} \leq K$. Furthermore, we assume that the *fresh* requests arrive according to a Poisson process with λ arrivals per subframe. Although we do not explicitly model the backlog mechanism, the *aggregate* random access requests, which comprises of both the fresh and the retransmitted requests, is assumed to form a Poisson stream[2] of rate a arrivals per subframe.

In Step 2, we consider that c uplink (UL) grants can be provided in a Message 2 which consumes N^{Msg2} Control Channel Elements (CCEs) in PDCCH. Since these messages/UL grants are not queued, they are lost if not sent in the timeslot immediately following the timeslot when Message 1 was sent because of small value of c. Failure to do so fails the random access process, and a backoff period begins before a retransmission attempt is made. Again, analogously as in Step 1, we denote by $Y_n^{(2)}$ and $\tilde{Y}_n^{(2)}$ the numbers of the *successful* and the *chosen* UL grants, respectively, sent in timeslot n. In a timeslot, at most bc UL grants can be sent. If $\tilde{Y}_{n-1}^{(1)} \leq bc$ then we do not have any loss in Step 2, i.e., $\tilde{Y}_n^{(2)} = \tilde{Y}_{n-1}^{(1)}$ and $Y_n^{(2)} = Y_{n-1}^{(1)}$. On the other hand, if $\tilde{Y}_{n-1}^{(1)} > bc$, then $\tilde{Y}_n^{(2)} = bc$ and the UL grants are sent corresponding to the randomly selected *chosen* preambles.

In Step 3, all the users that received a Message 2 corresponding to their chosen preamble send a Message 3. Thus, the number of *successful* Message 3 in timeslot n is $Y_n^{(3)} = Y_{n-1}^{(2)}$.

A Message 4 is generated by the eNB for every successful Message 3 received from the previous timeslot. Each Message 4 uses N^{Msg4} CCEs. These messages can be queued, and they should be received by the UE

[1]Note that this definition of a timeslot is different from the definition of a *time slot* described in the LTE standards and mentioned at the beginning of Section 2.3, which is always 0.5 ms long.

[2]The evidence for the validity of the Poisson assumption in provided in Appendix B of Publication I.

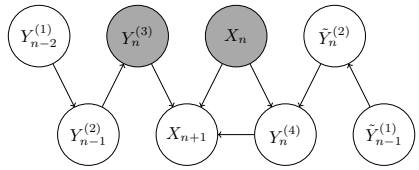


Figure 2.3. Random variable dependence graph for the variables in (2.1).

before the expiration of the retransmission timer. Otherwise the random access process fails. Let N be the size of PDCCH resource (in CCEs). Then a maximum of either $M = \frac{N}{N^{\mathrm{Msg4}}}$ or $m = \frac{N - N^{\mathrm{Msg2}}}{N^{\mathrm{Msg4}}}$ Message 4's can be sent in a subframe depending on the presence or absence of a Message 2 in the PDCCH. Additionally, if $c \leq m$, a Message 4 buffer need not be formed as there is enough capacity in the PDCCH to send a Message 4 corresponding to all the successful Message 3's. Under this condition, the process fails due to one of two reasons—either there is a collision in Step 1 or there is a loss in Step 2 because c is too small. On the other hand, if $c > m$, due to the limited capacity of PDCCH, a buffer may have to be used to keep Message 4's at the eNB before sending them. This may lead to the failure of the random access process, even if the two aforementioned failure events do not happen, due to a long buffering delay of Message 4.

With these assumptions, we observe that the Message 4 buffer evolves as

$$X_{n+1} = X_n - Y_n^{(4)} + Y_n^{(3)} = X_n - Y_n^{(4)} + Y_{n-1}^{(2)}, \text{ with} \tag{2.1}$$

$$Y_n^{(4)} = \min\{X_n, m\}\mathbf{1}_{\left\{\tilde{Y}_n^{(2)} > 0\right\}} + \min\{X_n, M\}\mathbf{1}_{\left\{\tilde{Y}_n^{(2)} = 0\right\}}, \tag{2.2}$$

where X_n and $Y_n^{(4)}$ denote the length of the Message 4 buffer and the number of Message 4's sent in timeslot n, respectively. The pair $\left(X_n, Y_n^{(3)}\right)$ is an aperiodic and irreducible Markov chain. The Markov property can be verified from the dependence tree of different variables as shown in Figure 2.3; aperiodicity and irreducibility follow from the construction of the chain.

Let the average throughputs of Message 1, Message 2 and the total number of uplink grants be denoted by $\theta^1(a), \theta^2(a)$ and $\tilde{\theta}^2(a)$, respectively. Furthermore, let the average leftover capacity for serving the Message 4 buffer per timeslot be denoted by $\sigma^{(4)}(a)$. In Publication I, § V, we describe in detail the method to calculate these quantities from the system parameter values. As the main theoretical contribution, we prove that the Message 4 buffer is stable if and only if

$$\theta^{(2)}(a) < \sigma^{(4)}(a)$$

in Appendix B of Publication I. That is, the Message 4 buffer is stable if and only if the throughput of Message 2's, which is sent in the PDCCH and possibly generate Message 4's, is less than the capacity of sending Message 4's in the PDCCH. Furthermore, let a_2^* and a_4^* represent the maximum arrival rate at which the Message 2's and Message 4's can be sustained, respectively, and $\theta^* = \theta^{(2)}(a_4^*)$ be the maximum throughput of the random access requests. Finally, we reiterate that the input parameter of our model is the aggregate arrival rate a. However, when the system is stable, the average arrival rate λ of the fresh requests is the same as the average throughput $\theta^{(2)}(a)$ of Message 2's. Therefore, the relation

$$\lambda(a) = \theta^{(2)}(a), \tag{2.3}$$

can be used to infer the arrival rate of fresh requests from the knowledge of the average throughput.

2.3.4 Numerical results

We explore the properties of the LTE random access process using the model described in Section 2.3.3. We focus mainly on calculation of the probabilities of various failure events and the overall throughput of the random access requests, when a large number of machines are present and PDCCH has a limited capacity.

The throughput of different messages for $c = 3$ and $c = 6$ are shown in Figure 2.4 as a function of a. Clearly, we observe that when $c = 3 = m$, curves $\sigma^{(4)}(a)$ and $\theta^{(2)}(a)$ do not intersect, no buffer is needed for Message 4's and the maximum throughput of the random access requests is the maximum throughput of Message 2's. On the other hand, when $m = 3 < c = 6$, $\sigma^{(4)}(a)$ and $\theta^{(2)}(a)$ intersect at the point that corresponds to maximum throughput θ^*, and a buffer for Message 4's is needed.

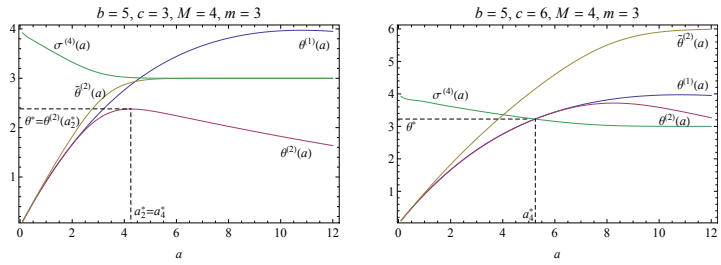


Figure 2.4. Throughput of various messages as a function of the aggregate arrival rate a when no queue for Message 4's is formed (left panel) and queue for Message 4 is formed (right panel).

We recall that the access attempts fail due to one of three reasons—collision, loss and long buffering delay. In Publication I, § VI.B we describe the methods to calculate the probabilities of the occurrence of the first two events. The probability of the occurrence of the third event is either zero when $c \leq m$, or when $c > m$, can be estimated using the simulation of the system dynamics as described in Publication I, § VII.C. The corresponding breakdown of the failure probability components are shown in Figure 2.5 where we clearly observe that Slotted Aloha is mostly responsible for random access failure.

By choosing different values of the parameters b, c, M and m we can generate various scenarios to study the behavior of the Message 4 buffer as shown in Publication I, Table 1. The buffer explodes only very close to the capacity limit, as seen in Figure 6 of Publication I, causing the probability of failure due to the delay to become more significant only very close to this stability limit (Figure 2.5, right panel).

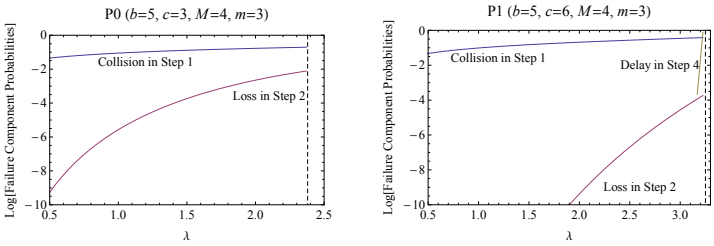


Figure 2.5. Different components of failure probability as a function of the fresh random access requests arrival rate λ when no queue for Message 4's is formed (left panel) and queue for Message 4 is formed (right panel). Note that for a stable system the arrival rate λ is the same as the access throughput $\theta^{(2)}(a)$ as mentioned in (2.3).

We also observe a clear dependence of the throughput and the stability of the random access requests on parameter c, the number of uplink grants sent in Message 2 from Figure 2.4. Increasing c necessitates the use of a buffer and raises the throughput of Message 2 and the random access requests. However, it is not always possible to raise the access throughput by increasing c indefinitely. The maximum throughput saturates after a certain value of c as the PDCCH is not able to accommodate more random access requests after a certain maximum throughput value (see Figure 4 in Publication I).

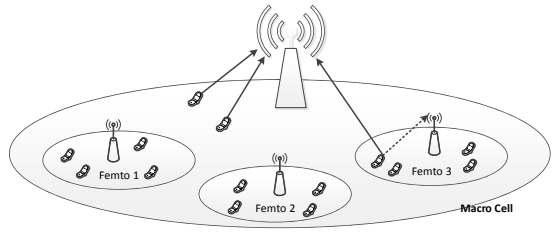


Figure 2.6. A heterogeneous network.

2.4 Load balancing in heterogeneous systems

From Section 2.3 we observe that a severe limitation on the operation of M2M devices is posed by the random access part of LTE. To overcome this limitation and sustain a large number of users in a cell, heterogeneous networks (HetNets) have been proposed [41], where inside the coverage area of a macrocell we have smaller femtocells serving the local traffic hotspots. They may be operating using the same radio technology (e.g., LTE), or the macro and the femto may use two different technologies (LTE in the macro and WLAN in the femto).

We assume that the femtos are the primary service providers. The macro handles the excess arrivals from the femtos, including its own background traffic as shown in Figure 2.6 and as studied in Publication II and Publication III for two different types of femtos. Each cell has its own backlog, which is the number of UEs whose access attempt has failed. We aim to select a part of fresh femto traffic to divert to the macro so that the aggregate mean backlog is minimized.

Let there be $m+1$ cells indexed by $i = 0, 1, 2, \ldots, m$ where $i = 0$ is the macro while the rest are the femtos, λ_i be the arrival rate of the random access requests in cell i and $X_i(t)$ be the backlog process in cell i. We assume that the femto cells are all homogeneous and have the same system parameters for operation, although the request arrival rates among them can be different. We consider just the first step of the random access process and not the data-transfer stage. This means that if the access to the network has multiple steps, as we have seen in the case of LTE in Section 2.3.3, we consider only the first step (i.e., for LTE it would mean the Slotted Aloha step where one of the preamble sequence is randomly selected and transmitted). Ignoring the later steps of the access process simplifies the model and allows us to extend the analysis to a multicell scenario.

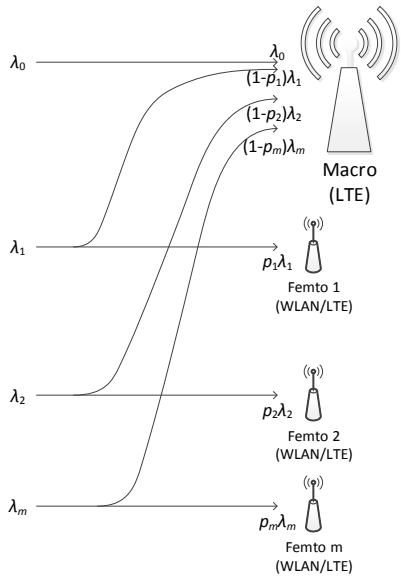


Figure 2.7. An illustration of the load-balancing problem.

The question we are interested to explore now is as follows: What is the optimal decision for a user coming to the femto—should it be served by the femto or the macro—so that the long-term mean aggregate backlog (and the access delay) is minimized? The decision is made at the arrival instant of the request. It can depend on the system parameters, which are assumed to be constant, and the time-varying instantaneous backlog (also called the *state*). However, once a user is assigned to a cell, it stays in the same cell until it is served, i.e., the backlogged UEs are not rerouted for retransmission. Mathematically, the optimization problem can be expressed as

$$\min \sum_{i=0}^{m} \mathbf{E}[X_i(t)],$$

where the minimization is done over all the *policies* for operating the cells. These policies, which take the parameters and the state of the system into account, decide whether a new user goes to the macro or stays at the femto cell (see Figure 2.7). We study the effect of applying several load-balancing policies in a heterogeneous wireless network, based on different heuristics that try to minimize the mean backlog and by Little's law, the mean access delay.

2.4.1 Stability issues

The load balancing is relevant only if all the cells are stable, i.e., the backlog of any of the cells does not explode. This is possible if and only if the service rate in each femto is larger than the arrival rate of random access requests there *after* balancing the load through the diversion of traffic. We consider cases where the excess traffic is sent to the macro from the femto. For stability, the macro must remain stable after the excess traffic from the femtos is sent to it. Therefore, the stability condition reads as

$$\lambda_0 + \sum_{i=1}^{m} \max\{\lambda_i - \mu, 0\} < \mu_0, \tag{2.4}$$

where μ and μ_0 represent the average service rates of the femtos and the macro, respectively.

2.4.2 Static load balancing

The static load balancing policies do not take the state of the system into account. For such policies, we assume that a fraction $0 < p_i < 1$ of traffic arriving to femtocell i at rate λ_i is kept in the cell and the rest of it, $(1 - p_i)\lambda_i$, is diverted to the macro. Under the assumptions of our model (see Sections 2.5.3 and 2.6.3) the average backlog length in each cell is a continuous and monotonously increasing function of the arrival rate of the access requests. Therefore, by suitably choosing p_i, we can minimize the backlog. This is equivalent to solving the optimizing problem

$$\underset{(p_1,\ldots,p_m)}{\arg\min}\, \bar{X}_0\big(\lambda_0 + \textstyle\sum_{i=1}^{m}(1-p_i)\lambda_i\big) + \displaystyle\sum_{i=1}^{m} \bar{X}_i(p_i\lambda_i), \tag{2.5}$$

$$\begin{aligned} \text{s.t.} \quad & 0 \le p_i \le 1, \\ & p_i\lambda_i < \mu, \qquad \forall i \in \{1,\ldots,m\}, \\ & \lambda_0 + \sum_{i=1}^{m}(1-p_i)\lambda_i < \mu_0. \end{aligned} \tag{2.6}$$

Note that in (2.5), $\bar{X}_i(\cdot)$ represents the average backlog in cell i which depends on the arrival rate of requests in that cell. The policy that solves the optimization problem (2.5) subject to the conditions (2.6) is called the *optimal static policy*.

In our studies, we use the optimal static policy as the baseline policy and attempt to devise other policies that can potentially outperform it.

2.4.3 Dynamic load balancing

In dynamic load balancing, we utilize the information about the state (i.e., the backlog) and possibly the parameter values of the system to assign UEs to a cell. A number of dynamic policies can be defined that can maintain maximal stability. We discuss a few of them and their characteristics.

MDP-PI-based policies
The random access of LTE and WLAN can be modeled using Markov Chains and Markov Processes, respectively (see Sections 2.5.3 and 2.6.3). Assigning a new UE in the macro or the femto affects the future evolution of the backlog process in both the macro and the femto. This process can be optimized, i.e., the cost or the average backlog can be minimized, using the Policy Iteration (PI) algorithm from the theory of Markov Decision Processes (MDP) [70]. In this technique, for a discrete-time system, starting from the initial policy π we choose an action a_i in state i such that

$$a_i = \arg\max_a \left\{ r_i(a) - r(\pi) + \sum_j p_{i,j}(a) v_j(\pi) \right\}.$$

Here the terms inside the braces is the future cost of the system if action a is chosen at current decision instant and the policy π is used thereafter, $r_i(a)$ is the immediate cost of the current state with action a, $r(\pi)$ is the average cost with initial policy π, $p_{i,j}(a)$ is the transition probability from state i to j for action a, and $v_i(\pi)$ is the relative value of state i which is the difference of the average cost of a process starting in state i and a stationary process under policy π. The relative values of states can be calculated by solving the so-called Howard equations which is a set of linear equation obtained for each state. The solutions of the Howard equations also provide the average cost of policy π.

By choosing an action a_i associated with each state i, we obtain a new policy π' which is guaranteed to be not worse than the initial policy π. The PI can be further continued to get better policies until the convergence to the optimal policy is achieved. Typically, policy iterations converge rapidly (within a few iterations). For continuous time systems, an analogous formulation can be devised, which is presented in Section 3.3.5.

Moreover, when we begin from the optimal static policy, all the backlog processes in the cells are independent of each other. Therefore, we have a combination of $m+1$ one-dimensional processes to optimize. After the first iteration, we obtain an $(m+1)$-dimensional process and the complete

policy iteration is numerically feasible to perform only for small values of m. In many problems, however, it has been observed that the first iterated policy is close to the optimal policy.

JSQ-based policies
Join shortest queue (JSQ) is a load-balancing heuristic which suggests that each incoming user should choose the station that has the shortest backlog. It is a very intuitive policy, and in the task assignment problems for parallel servers, it has been shown to be optimal under certain conditions. It should be noted that our problems, which are similar to the load balancing problem using an overflow channel, are qualitatively different from such dynamic task assignment problems, where the load is balanced by properly selecting any of the available servers. However, a *suitable* JSQ policy and its variants can nevertheless be applied. So, with JSQ a user arriving at time t is assigned to the femto cell i if and only if

$$X_i(t) \leq X_0(t). \tag{2.7}$$

Besides these broad classes of dynamic policies, various other dynamic policies can be defined based on other heuristics and the knowledge of the behavior of the system with the information of instantaneous state and system parameters. They are discussed in detail in Sections 2.5 and 2.6, where we study the system-specific load balancing problems.

2.5 LTE-LTE HetNet load balancing

2.5.1 Research problem

The load balancing principles discussed in Section 2.4 are studied in Publication II in the context of a network that has LTE cells for both the macro and the femto, operating independently of each other. This allows the backlogs in all cells to be modeled by MCSA, as discussed below in Section 2.5.3. The aim again is to minimize the average backlog length and thereby average access delay using various dynamic policies.

2.5.2 Multichannel Slotted Aloha

Multichannel Slotted Aloha (MCSA) is the Slotted Aloha protocol where multiple parallel channels are available for data transmission. It is used

as the first part of the random access process in systems such as LTE. The user selects one of K available channels and attempts to transmit data in it. In LTE, a channel corresponds to a preamble. If no other user selects the same preamble at the same time, then the user is said to be successful. Otherwise, if two or more users select the same channel at the same time, then a collision occurs and both the users are said to be backlogged. The backlogged users may attempt a retransmission, in a randomly selected channel after some time called the backoff duration. To overcome the inherent instability of the protocol, the backoff duration of a request typically increases after every collision. Moreover, a user makes some maximum number of retransmission attempts, after which the request is dropped if the collision still persists.

2.5.3 MCSA model

We model the evolution of backlog in a MCSA by a Markov chain as explained in Publication II and Publication III. We assume that there are K channels (preambles). The duration of one random access timeslot is denoted by τ for the Slotted Aloha. Let A_n denote the number of access requests arriving during timeslot n and D_n denote the number of non-colliding requests. The backlog X_{n+1} at the beginning of timeslot $n+1$ evolves as

$$X_{n+1} = X_n + A_n - D_n, \tag{2.8}$$

which is a Markov chain. Additionally, we assume that A_n is Poisson distributed with mean $\lambda\tau$.

The state-dependent retransmission probability of a backlogged request is denoted by $r_n = r(X_n)$, which models the backoff time of a backlogged user. It assumes that the backlog mechanism is aware of the total backlog. A state-dependent retransmission probability that keeps the system stable should be devised because the chain (2.8) is unstable when r_n is a constant [53]. In Publication II and Publication III, we choose $r(n)$ as

$$r(n) = \min\left\{\frac{K - \lambda\tau}{n - \lambda\tau/K}, 1\right\},$$

which also ensures that the system is stable whenever

$$\lambda < K/(\tau e). \tag{2.9}$$

Furthermore, we assume that both the arriving users and the retransmitting users independently and randomly select one of the K available

preambles. The state transition probability from state i to j is then

$$p_{ij} = \sum_{k=0}^{K+j-i} \frac{(\lambda\tau)}{k!} e^{-\lambda\tau} \sum_{\ell=0}^{i} p(K, k+l, i+k-j) \binom{i}{\ell} r(i)^i (1-r(i))^{(i-\ell)},$$

where $p(K, n, m)$ is the probability to assign n balls into K boxes so that m of them have exactly one ball, and is calculated in Publication II. From this state transition probabilities, the steady state probabilities and the mean backlog $\bar{X}$ readily follow.

2.5.4 Operating policies

Now, we discuss the various policies that we use to operate the cells in the case when both the macro and the femtos are based on the MCSA protocol. Since all the cells are homogeneous, we have $\mu_0 = \mu = K/(e\tau)$, in the stability condition (2.4), which is the maximum arrival rate for which a MCSA cell with K preambles can be stabilized as seen from (2.9).

Optimal static policy
As discussed in Section 2.4.2, it is possible to find the optimal probability vector $(p_1, \ldots, p_m)$ that minimizes the total mean backlog. In the Appendix of Publication II, we give an efficient algorithm to arrive at a solution of this optimization problem. For this particular case with symmetric cells, the static optimal policy is quite intuitive—it attempts to make the load as equal as possible in all the cells. This policy is used as the baseline policy for comparison with other policies.

Min-max policy
The Min-max policy attempts to mimic the JSQ-heuristic and minimize the maximum number of users (fresh and backlogged) in all the cells in every timeslot by diverting the fresh femto arrivals to the macro. We propose a 'reverse-waterfilling' algorithm to do this allocation assuming that we know the backlogs in all the cells at the beginning of the timeslot and the number of arrivals during the timeslot. Initially we make the allocations to the respective cells, and then from the femto with the largest backlog we put new arrivals to the macro, one at a time, until macro's potential backlog is larger than the femto's potential backlog. This is a very simple approach and takes at most $\sum_{i=1}^{m} a_i$ steps where a_i is the number of new arrivals in cell i. It should be noted that such allocation may not be unique. The details of the algorithm are discussed in Publication II, § IV.B.

Max-throughput policy
In the Max-throughput policy, using the knowledge of instantaneous backlog and the number of arrivals during a timeslot, we attempt to make allocations such that the total immediate random access throughput in all cells is maximized. All the possible allocations are exhaustively tried to find the one that gives the maximum possible throughput in all cells as discussed in Publication II, § IV.C.

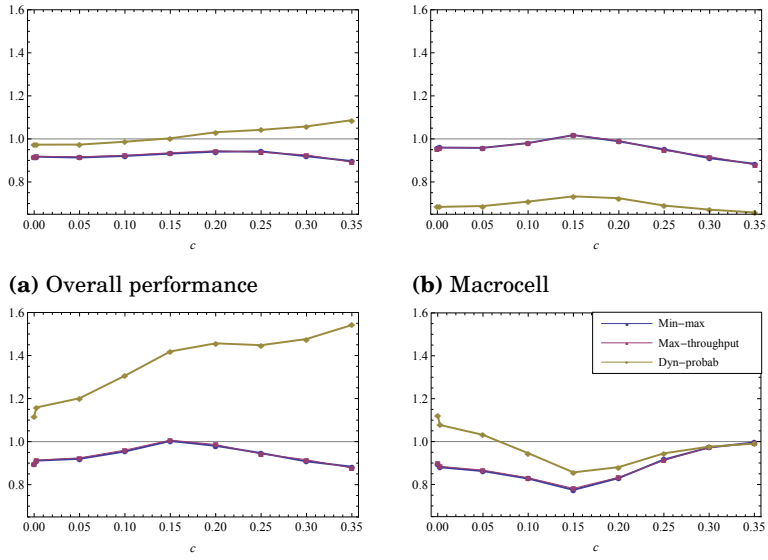
Dyn-prob policy
The Dyn-prob policy uses only the information about the backlogs at the beginning of the timeslots in all the cells to calculate the probabilities $\tilde{p}_i$ of keeping an incoming user in femtocell i at the same femto. It is robust as it does not need to estimate values of the arrival rate and the system parameters. This policy is inspired from the algorithm for calculating the optimal static policy and described in Publication II, § IV.D.

MDP-based policy
As discussed in Section 2.4.3, MDP based policy iteration is performed to determine the optimal policy. The details about determining the value functions and the cost functions are discussed in Publication II, § V. For a small system ($m = 1$), we can perform the full policy iteration, while for a larger system ($m > 1$), only the first policy iteration is numerically feasible. It provides an estimate of the performance bound.

To implement the optimal static policy, only the arrival rates and system parameter values at different cells need to be estimated. Among the dynamic policies, only the robust Dyn-prob policy is *implementable* as it does not need the information about the future arrivals. The other dynamic policies need, in addition to the state information, the prediction from an *oracle* about future arrivals to make allocation and possibly an estimate of the arrival rate and system parameters.

2.5.5 Numerical results for MCSA-based femtos

We study through simulations different scenarios to gain an insight into the performance of various policies in our system. A complete list of scenarios and assumptions is presented in Publication II, § VI. A general observation from the scenarios is that the dynamic policies that use all the possible information (Max-throughput, MDP) can improve the performance over the static optimal policy especially at lighter loads and when



(a) Overall performance

(b) Macrocell

(c) Heavily loaded femtocell (Femto 1)

(d) Lightly loaded femtocell (Femto 2)

Figure 2.8. Scenario 3 in Publication II: Performance ratios of the dynamic policies with two femtocells ($m = 2$) having asymmetric arrival rates for the overall mean backlog (top, left), macrocell (top, right), heavily loaded femtocell (bottom, left) and lightly loaded femtocell (bottom, right). Note that the Min-max and Max-throughput curves overlap here.

the macro is not loaded. However, even the robust Dyn-prob policy appears stable and performs as good as the optimal static policy when the macro and the femtos are equally loaded. Moreover we can also speculate that Min-max, which is computationally feasible even for a large system and does not need any information about the arrival rates, is performing very close to the optimal policy and can be used as an estimate of the optimal performance.

Here we show one of the asymmetric scenarios (Scenario 3 in Publication II) for reference. We choose $m = 2$ and $\lambda_0 = K\tau/(2e)$, and the arrival rates in the femtos are chosen as $\lambda_1 = (1 + c)K/(\tau e)$ and $\lambda_2 = (1 - c)K/(\tau e)$, where $c \in (0, 0.5)$ so that the system is always stable. We plot the ratio of the total mean backlog with different policies to the same with the optimal static policy as a function of parameter c. The dynamic policies except Dyn-prob consistently provide a gain of 10% or more as seen in Figure 2.8a. The robust Dyn-prob performs very close to the optimal static policy. In fact, in the macro (Figure 2.8b) Dyn-prob is the best performing policy, while in the lightly loaded femto (Figure 2.8d) it is performing very badly compared to the others as well as the optimal static policy.

With high values of c, the performance of all the dynamic policies become worse at the lightly loaded femto, while in the heavily loaded femto the performance of the near-optimal dynamic policies improve (Figure 2.8c).

2.6 LTE-WLAN HetNet load balancing

2.6.1 Research problem

The load balancing policies discussed in Section 2.4 are studied in Publication III for a network that has LTE cells in the macro and the WLANs in the femto, operating independently of each other. This means that now the backlog of the macro can be modeled by MCSA, as discussed in Section 2.5.3, while the Markov model for the femtos is provided in Section 2.6.3. The aim again is to minimize the average backlog and thereby average access delay using various dynamic policies.

2.6.2 CSMA protocol

The femtos are WLAN based and thus use the Carrier Sense Multiple Access (CSMA) protocol for data transmission. CSMA is a media access control (MAC) protocol in which a user first makes sure that the channel is free (i.e., no other user is transmitting data using the same medium) and only after that it transmits its own data to avoid collision. The channel is said to be busy when any user is transmitting its data. Otherwise it is said to be idle. When a user senses channel as busy, it backs off and waits some random time before sensing the channel. Only when the channel is sensed as idle, it transmits the data. The sensing of CSMA is said to be *perfect* or *ideal* if it can detect a transmission by other users immediately, i.e., with no propagation delay.

2.6.3 CSMA model

When the femtocells are based on the CSMA protocol, we use a continuous time Markov model similarly as done in [17, 22, 20] to study the behavior of the femtos.

We assume that the random access requests arrive at the femto according to a Poisson process with rate λ. If the channel is sensed idle at arrival, then the packet transmission begins immediately. This time is assumed to

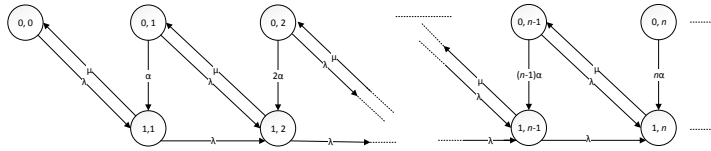


Figure 2.9. The Markov model for the CSMA in WLAN used in Publication III.

be exponentially distributed with mean $1/\mu$. If the channel is in the busy state and a new arrival occurs, the user is backlogged for a time which is exponentially distributed with mean $1/\alpha$. After the backlog timer expires, the user transmits if it senses the channel as free. Otherwise, a new backlog phase with exponentially distributed time with the same mean $1/\alpha$ is started. We assume that the UEs are so close to the base station that the sensing is perfect and there are no collisions.

Let $X(t)$ denote the total number of users (backlogged as well as the one in transmission) at time t and $B(t)$ the state of the channel which is 1 or 0 depending on whether the channel is busy or free, respectively. The pair $(X(t), B(t))$ is a Markov process whose state-transition diagram is shown in Figure 2.9, from which the steady-state probabilities, $\pi(x, b)$, and the mean number of users, $\bar{X}$, are calculated as

$$\pi(x,b) = \left(\frac{\lambda}{\mu}\right)^x \left(1-\frac{\lambda}{\mu}\right)^{1+\frac{\lambda}{\alpha}} \prod_{y=1}^{x-1} \left(1+\frac{\lambda}{y\alpha}\right) \left(\frac{\lambda}{x\alpha}\right)^{1-b} \text{ and} \tag{2.10}$$

$$\bar{X} = \sum_{x=1}^{\infty} x(\pi(x,0)+\pi(x,1)) = \frac{\lambda\left(1+\frac{\lambda}{\alpha}\right)}{\mu-\lambda}, \tag{2.11}$$

respectively, whenever the chain is stable, i.e.,

$$\lambda < \mu.$$

2.6.4 Operating policies

We now discuss various policies we have used to balance the load from the femtos to the macro. Note that the overall stability condition (2.4) should still be satisfied with $\mu_0 = K/(e\tau)$, which is the maximum arrival rate for which the macro with K preambles can be stabilized as seen from (2.9).

Optimal static policy

As in the case of MCSA-based femtos, we can define the static and the optimal static policies in a similar way. The optimization problem is now more difficult than that for the case with LTE in both macro and the fem-

tos, which is described in the Appendix of Publication III. The basic principle of the algorithm is that due to symmetry of the femtos, for any amount of overflow traffic in the macro, it is possible to easily determine the optimal share of the traffic from each femto. This reduces the m-dimensional optimization problem to a one-dimensional optimization problem of finding the optimal total overflow traffic in the macro. Again, this policy is used as the baseline policy for comparing the other dynamic policies.

JSQ-based policies

With JSQ a user arriving at time t is assigned to the femto cell i if and only if it does not have longer backlog than the macro, i.e., (2.7) is satisfied. To account for the difference in the service capacities of the macro and the femtos, the JSQ policy can be modified to the JSQ-μ policy which selects the femto cell i if and only if

$$\frac{X_i(t)}{\mu} \leq \frac{X_0(t)\tau e}{K}.$$

Finally the improved version of JSQ called the JSQ-I, will always put a new user to the femto if it senses the channel as free. If the channel is busy, the user is assigned to the femto if the JSQ heuristic (2.7) is true. Otherwise it is put in the femto.

We note that in all the JSQ-based policies, the ties are broken in the favor of the femtos, which operate in continuous time and where service can possibly begin immediately at arrival.

FPI policy

We can define the MDP based policy for this system as well. The details for determining the value and the cost functions are provided in Publication III, § 3.3. Note that we begin from the optimal static policy and perform only the first policy iteration to get a new policy. Performing further iterations even for a small system is restricted by the fact that we are dealing with mixed system with discrete and continuous time elements.

Again, we note that the dynamic policies mentioned above are *non-implementable* as they utilize the information about the access requests arriving in future from an *oracle* and possibly an estimate of the access request arrival rates and the system parameters. Only the JSQ-I policy is *implementable* as it needs only the state information.

2.6.5 Numerical results for CSMA-based femtos

We again use simulations to explore the dependence of the access delay on different policies. Due to computational limitations, we restrict our study to $K = 20$ preambles. We construct various scenarios in Publication III by changing different system parameters. A selection of these results is presented here (for Scenarios 1–3 in Publication III). Scenario 1 is the basic scenario with two femtos ($m = 2$), $\mu = K/(\tau e)$ and $\alpha = \mu$. Scenarios 2 and 3 have femtos with half and double capacities, respectively, compared to the femtos in Scenario 1. The performance is measured as the ratio of the total average backlog in all the cells using the dynamic policy to the same using the optimal static policy, which by Little's result, is the same as the ratio of access delays using the same two policies. These ratios are plotted in Figure 2.10 as a function of normalized load.

A clear message from these results is that FPI-based policy is outperforming all the other policies, and as expected, is always better than the optimal static policy. As the policy iteration algorithm is known to rapidly converge to the optimal policy, we can speculate that the results from the FPI policy are close to the optimal results. Moreover, the JSQ-based policies which are far simpler and use much less information, in general, than FPI, are also showing good performance.

In Scenario 1 (Figure 2.10, left panel), we see that using a JSQ-based policy, it is always possible to get performance gain at relatively high loads and it can be as much as 30% better than the optimal static policy. Since both the macro and femto have the same capacities, JSQ and JSQ-μ are identical here. When the service rate of the femtos is halved, JSQ-μ is consistently performing better than the other two JSQ based policies (Figure 2.10, center panel) and the optimal static policy. When we double the service rate of the femtos compared to that of the macro (Scenario 3, Figure 2.10, right panel), we have gains only at the high values of the load, and now JSQ-I is the best performing JSQ-based policy. Moreover, the service-completion rate aware JSQ-μ shows the worst performance at higher loads. When we look at the pictures of the absolute backlog in this scenario, as shown in Figure 2.11, we observe that at high loads JSQ-μ tends to keep users at the femtos and as a consequence, the performance of the whole system suffers. Clearly, modeling the heterogeneity of the system by just taking into account the service capacities of the macro and the femtos does not always work and may even prove to be

counterproductive in certain occasions.

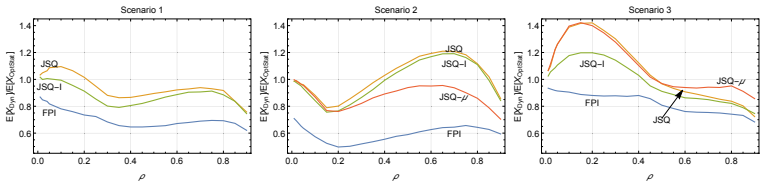


Figure 2.10. Ratio of total backlog using the dynamic policy to backlog using the optimal static policy for Scenario 1 (left), Scenario 2 (center) and Scenario 3 (right) in Publication III.

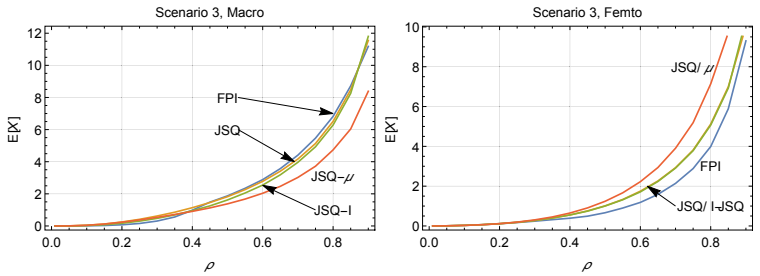


Figure 2.11. Cell specific backlog for Scenario 3 in Publication III. The left panel show the backlog of the macro while the right panel shows the backlog of the femto.

2.7 Summary

In this chapter, we have described the issues related to the random access in a modern LTE system especially in the context of M2M communication. First we saw that the signaling channel PDCCH acts as a bottleneck when a large number of devices attempt to enter the network. The model we present is useful for analyzing this bottleneck. We are able to determine the maximum throughput of the random access requests and even determine the probability of failure of these requests due to different reasons.

Next we looked at how the congestion in M2M random access be alleviated using a heterogeneous network. We studied two types of heterogeneous networks—LTE-LTE and LTE-WLAN—HetNets and devised various load balancing policies that help to minimize the access delay to the network. Our results show that the dynamic policies that take into account the current backlog do perform better than the static policies in most of the cases.

3. Intercell coordination in wireless systems

3.1 Introduction

The signal transmitted by a base station or any of its users appears as noise to the users of the other cells. Thus, the base station and the UEs present in one cell can adversely affect the performance of the other cells due to what is called *intercell interference*. With the increasingly denser deployment of wireless networks, the issue of intercell interference is one of the most important problems that needs to be addressed at the access network.

By taking into account the operation of neighboring base stations and the actions of the users present there, the eNBs can collaborate with each other to combat the intercell interference and provide better services to their UEs. The harmful interference from neighboring stations can therefore be mitigated through proper coordination of transmission power levels among the eNBs. With the development of technologies like CRAN, it is even possible to implement coordination schemes that are centrally controlled. Such schemes generally have a more global view of the network. Thus, they are able to exploit the available information about the arrival rates, the service rates, the congestion levels, etc., more effectively than the distributed schemes which have only local information, ultimately leading to better performance.

The traffic that we consider here are called *flows*. They are elastic, which permits considerable variation in the service rate during the flow's transmission, allowing less stringent constraints for operation. The dynamics of such elastic traffic are better manifested at the flow level. Assuming the timeslots are fairly shared among the flows and time-scale between flow-level dynamics and timeslot-level channel dynamic are separated by

a few orders of magnitude, we can use PS queues to model the service of the base stations.

Moreover, there may be various categories of users in a cell that have service demands of different nature. These are modeled by using multiple PS queues with independent arrivals, and, where applicable, assigning different classes to various users in the same queue. Thus, we have a system with multiple *interacting* queues where the coordination scheme affects the service rates of the queues in a complex way. Such coordination schemes, in effect, lead to the selection of the service rates of the different queues from the so-called abstract *capacity region*. The geometry of the capacity region depends on the number of queues and the nature of coupling of the service rates of the queues in the different operation modes.

In this chapter, we study the problem of coordination of two interfering base stations so that the flow-level delay is minimized. We consider just two neighboring cells because it leads to models that are tractable to analysis, are not too computationally taxing even for simulation-based studies, and at the same time provide useful results relevant to the real systems as well.

To this end, we begin by noting that in a single-class system with symmetric service rates, the policy that uses both the stations as much as possible for downlink transmissions has a very strong optimality property when the interference is low [95]. Thus, we investigate the effectiveness of similar schemes when there are multiple location-based classes of users in a cell, as presented in Publication IV. This scheme is tested against a policy developed using the techniques from the theory of Markov Decision Processes (MDP), which systematically improves a given baseline policy.

In Publication V and Publication VI, we study the coordination in a dynamic TDD system, where the fraction of timeslots given to UL and DL can dynamically vary depending on the load conditions. The problem is then formulated as a dynamic control problem of operating four interacting queues. We study various static and dynamic policies of operating these stations so that the average queue length and flow-level delay is minimized, leading to higher system throughput.

3.2 Related work

The exponential increase of data demand and the need for dense deployment of cells is discussed in [41], and the solutions offered by CRAN is

also discussed in detail in [39]. The challenges related to base stations coordination is a classic problem in cellular systems and is well studied in [65, 85].

The flow-level models and PS queues are generally used to study the performance in wireless system. They are mostly discussed for either the uplink or the downlink in a single cell in [5, 25, 28, 29, 43, 94]. The issues of jointly controlling two or more coupled queues in different cells is more complex, and in works such as [29, 30, 60], the stability of such coordination problems are studied.

The literature related to the optimization in interacting queueing system is scarce. For two parallel queues, that includes independent as well as interacting servers, an optimal control in the form of a switching curve in two dimension has been derived in [47] that minimizes the average delay. In [95], the authors have been able to derive a stochastically optimal policy for a system with two interacting processor-sharing queues. This policy always serves with both the stations if there are users in both of them. They, however, consider single-class users and symmetric service rates, and such a stochastically optimal policy for multiple user classes is not known in general.

For a case of a cellular system with multiple base stations with sectors facing in different direction, [73, 74] discuss optimal scheduling rules for the users. The multiple users classes in the cells and the different transmission power levels of the stations are modeled using interacting queues. Processor sharing models have been used to study different static and dynamic scheduling rules in this context.

The dynamic TDD system has been discussed in [85], and the strict power control necessary for its operation is mentioned in [64]. The timeslot-level resource allocation in dynamic TDD is studied in [36, 87]. In [56], the optimization of TDD scheme in single cell is presented at the flow level. For multiple cells, dynamic TDD resource allocation is analyzed in [9] at the timeslot level.

In this chapter, we study various flow-level scheduling rules that can be jointly applied to neighboring base stations. The Max-Weight rule studied here is introduced in [92]. It has been shown to be maximally stable [92] and asymptotically optimal for interacting queues in discrete time [86] with a fixed number of flows.

Another scheduling scheme we study here is the Balanced Fair rule, which is insensitive to the distribution of the service time of the flows.

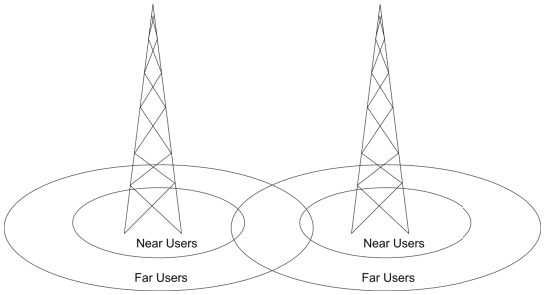


Figure 3.1. Interfering base stations with near and far users.

It is introduced in the context of fixed networks in [23, 24]. It has been expanded to include wireless networks in [68].

The theory of Markov Decision Processes gives a well known method for optimizing the scheduling decision when the flow service times are exponentially distributed. In particular, the policy iteration algorithm [70] that we use here has been used in a wide variety of contexts for optimizing scheduling decisions. To use this algorithm, the relative values of state for each state should be known [48], which, in general do not have a closed form expression, and need to be calculated numerically in many contexts. We use a technique provided in [58] to determine the value functions of a multi-class PS queue, which allows to carry out the optimization.

3.3 Coordination with location-based user classes

We start with the coordination problem in a system with two interfering base stations, which serve the users according to their location in the cell. The study of this problem is the main topic of Publication IV.

3.3.1 Research problem

We take a case where there are two neighboring base stations, where the users are classified according to their locations. The users can either be near to the station or far from it, as shown in Figure 3.1. The stations serve the users in a fair way, e.g., by using round-robin scheduling. However, because of the path loss effects, the near users are served at a higher rate than the far users.

For a single class of users, it has been shown in [95] that when the base stations have the same service rates (*symmetric*), and the interference is

low[1], it is stochastically optimal[2] to serve with both the stations if there are users in both the cells, and use only one station when the other has no user. However, when we have two classes of users in each station, this strategy does not always simultaneously maximize the service rate and departure rate of the flows. This can occur, e.g., when one cell has only far users and the other has only near users as explained in Section 3.3.5. Thus, the strong optimality property of this policy may be lost, although it may still show good performance. We explore the performance of the "Both Stations On" policy in a setting with two user classes in each cell, and compare it with an MDP-based policy.

3.3.2 System model

We consider the downlink operation of two neighboring base stations, labeled 1 and 2, respectively. Both stations, in their respective cells,[3] serve the elastic flows[4] of the downlink users that are near (n) to the station or far (f) from it. Thus, there are four classes of flows, which are identified by their location $l \in \{\mathrm{n}, \mathrm{f}\}$, and the station $i \in \{1, 2\}$ they are attached to. In each class (l, i), the flows arrive according to independent Poisson processes of rate λ_i^l, which are the components of $\boldsymbol{\lambda} = (\lambda_1^{\mathrm{n}}, \lambda_1^{\mathrm{f}}, \lambda_2^{\mathrm{n}}, \lambda_2^{\mathrm{f}})$. The flows in class (l, i) have random sizes B_i^l (bits), which are assumed to be independent and identically distributed with an arbitrary distribution. Furthermore, since each station can either be turned on for downlink or turned off, there are three operating modes $m \in \{0\mathrm{d}, \mathrm{d}0, \mathrm{dd}\}$[5], when at least one of the stations is serving. The first letter of the mode denotes the operating state of Station 1, while the second letter denotes the operating state of Station 2. For example, $m = 0\mathrm{d}$ indicates that Station 1 is turned off, while Station 2 is turned on for downlink.

Let $r_i^{m,l}$ denote the service rate (in bits per second) received by a user of class (l, i) when operated in mode m if it is the only user present in the system and $\mu_i^{m,l}$ denote the corresponding average service completion rate defined as

$$\mu_i^{m,l} = \frac{r_i^{m,l}}{\mathbf{E}[B_i^l]}.$$

When more users are present, the station provides equal share of time to

[1]The *low interference* condition is defined precisely in Section 3.3.2.
[2]The definition of *stochastic optimality* is provided in Section 3.4.6.
[3]'Base station', 'cell' and 'station' are used interchangeably in this chapter.
[4]'Flow' and 'user' are used interchangeably in this chapter.
[5]Note that there can be an additional '00' mode in which both the stations are turned off. But, since the system is non-idling this mode is not used at all.

all the users associated with it. Furthermore, we define

$$k_i^m = \frac{\mu_i^{m,\mathrm{f}}}{\mu_i^{m,\mathrm{n}}},$$

such that $k_i^m < 1$ for all modes m, i.e., the far users are served at a slower rate than the near users in any mode by both stations.

Furthermore, we focus only in the scenario where the service rates of the stations are identical and k_i^m is constant. This *service rate symmetry* between the stations in every mode implies that

$$\mu_1^{\mathrm{d0,n}} = \mu_2^{\mathrm{0d,n}} = \mu_0, \quad \mu_1^{\mathrm{dd,n}} = \mu_2^{\mathrm{dd,n}} = \mu_1, \quad k_i^m = k$$
$$\mu_1^{\mathrm{d0,f}} = \mu_2^{\mathrm{0d,f}} = k\mu_0, \quad \mu_1^{\mathrm{dd,f}} = \mu_2^{\mathrm{dd,f}} = k\mu_1,$$

and obviously, $\mu_1^{\mathrm{0d,n}} = \mu_2^{\mathrm{d0,n}} = \mu_1^{\mathrm{0d,f}} = \mu_2^{\mathrm{d0,f}} = 0$, as there is no service when a station is turned off. The service rate of a station when the other is turned on is lower than when it is turned off, i.e.,

$$\mu_1 < \mu_0.$$

We also assume the aggregate service rate when both stations are turned on is higher than the same when only one is turned on, i.e.,

$$\mu_1 < \mu_0 \leq 2\mu_1,$$

which is also called the *low interference* condition and is depicted in Figure 3.2 (left panel).

We denote the instantaneous state of the system at time t by vector $\boldsymbol{X}(t) = (X_1^{\mathrm{n}}(t), X_1^{\mathrm{f}}(t), X_2^{\mathrm{n}}(t), X_2^{\mathrm{f}}(t))$ where $X_i^l(t)$ is the number of flows in class (l, i). Within each station, the flows are fairly served, e.g., in round-robin manner, whereby each flow receives equal time share from the station, irrespective of their location in the cell, at the timeslot level. With the additional assumption of time-scale separation between the scheduling time and the flow-level dynamics, we can say that the users are served according to the two-class PS queueing discipline [28]. Thus, we have four flow classes with independent arrivals, served in two queues with coupled service rates, represented by different operating modes.

3.3.3 Optimal coordination problem

The optimal coordination problem is to select the different modes of the stations so that the average flow-level delay is minimized, using information such the arrival rates, the service rates and queue lengths. We

begin by describing a class of *static coordination policies* and then, in Section 3.3.5 we discuss the *dynamic policies*.

A *static policy* is defined by the vector $\boldsymbol{p} = (p^{\mathrm{d0}}, p^{\mathrm{0d}}, p^{\mathrm{dd}})$ whose components p^m represent the probability that mode m is used to operate the base stations in a timeslot. They can also be interpreted as the time fractions the different modes are used in the long run. Thus,

$$0 \leq \boldsymbol{p} \leq 1 \quad \text{and} \tag{3.1}$$

$$|\boldsymbol{p}| \leq 1, \tag{3.2}$$

where the vector operations are defined as follows: For a vector $\boldsymbol{v}$, $|\boldsymbol{v}|$ denotes the sum of the components of $\boldsymbol{v}$, while an inequality involving a vector and a scalar means that the inequalities are to be taken componentwise in relation to the scalar. Note the strict inequality in (3.2) can occur if the system is idling, i.e., the mode '00' can also be used in addition to the three modes described in Section 3.3.2. The equality holds for a non-idling system.

Since $\boldsymbol{p}$ is constant, the two PS queues in the two stations are no longer coupled, and thus, we get two independent parallel PS queues, each with two classes of location-based users. The service rate ϕ_i of Station i is then given by

$$\phi_1 = p^{\mathrm{d0}}\mu_0 + p^{\mathrm{dd}}\mu_1 \quad \text{and} \quad \phi_2 = p^{\mathrm{0d}}\mu_0 + p^{\mathrm{dd}}\mu_1, \tag{3.3}$$

which comprise the components of $\boldsymbol{\phi} = (\phi_1, \phi_2)$. Therefore, the total average queue length is

$$\mathbf{E}[|\boldsymbol{X}|] = \frac{\lambda_1^{\mathrm{n}} + \lambda_1^{\mathrm{f}}/k}{p^{\mathrm{d0}}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_1^{\mathrm{n}} - \lambda_1^{\mathrm{f}}/k} + \frac{\lambda_2^{\mathrm{n}} + \lambda_2^{\mathrm{f}}/k}{p^{\mathrm{0d}}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_2^{\mathrm{n}} - \lambda_1^{\mathrm{f}}/k}. \tag{3.4}$$

The set of all service rates $\boldsymbol{\phi}$ that satisfy (3.1), (3.2) and (3.3) is called the *capacity region* and is denoted by $\mathcal{C}$. In this case, the capacity region can be explicitly characterized by the inequalities

$$0 \leq \phi_1 \leq \mu_0 - \frac{\mu_0 - \mu_1}{\mu_1}\phi_2 \quad \text{and} \quad 0 \leq \phi_2 \leq \mu_0 - \frac{\mu_0 - \mu_1}{\mu_1}\phi_1, \tag{3.5}$$

which is also depicted in Figure 3.2.

The *optimal static policy* is obtained by properly selecting the components of $\boldsymbol{p}$ that minimizes the average number of flows calculated in (3.4),

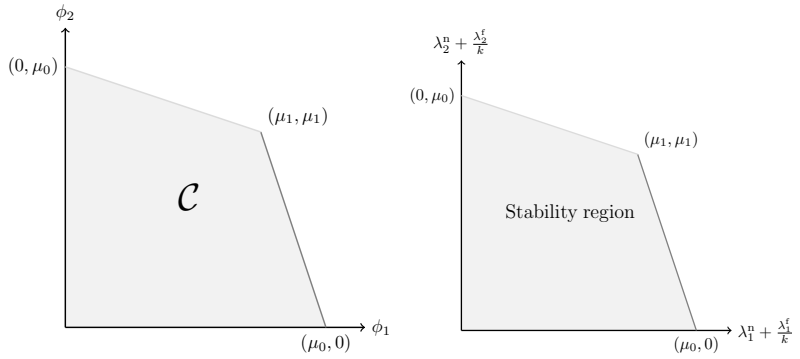


Figure 3.2. The left panel shows the feasible rate region $\mathcal{C}$ while the right panel shows the stability region of the model described in Section 3.3.2.

and by Little's result the average flow-level delay, as well, i.e.,

$$\begin{aligned}
\boldsymbol{p}^* = \underset{(p^{\mathrm{d}0}, p^{0\mathrm{d}}, p^{\mathrm{dd}})}{\arg\min} \Bigg\{ & \frac{\lambda_1^{\mathrm{n}} + \lambda_1^{\mathrm{f}}/k}{p^{\mathrm{d}0}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_1^{\mathrm{n}} - \lambda_1^{\mathrm{f}}/k} \\
& + \frac{\lambda_2^{\mathrm{n}} + \lambda_2^{\mathrm{f}}/k}{p^{0\mathrm{d}}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_2^{\mathrm{n}} - \lambda_1^{\mathrm{f}}/k} \Bigg\}, \\
\text{s.t.} \quad & 0 \leq \boldsymbol{p} \leq 1, \quad |\boldsymbol{p}| \leq 1, \\
& p^{\mathrm{d}0}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_1^{\mathrm{n}} - \lambda_1^{\mathrm{f}}/k < 0, \\
& p^{0\mathrm{d}}\mu_0 + p^{\mathrm{dd}}\mu_1 - \lambda_2^{\mathrm{n}} - \lambda_2^{\mathrm{f}}/k < 0,
\end{aligned}$$

which is a convex optimization problem. Let $\boldsymbol{\phi}^* = (\phi_1^*, \phi_2^*)$ be the allocation of the service rate of the optimal static policy. In Publication IV, the optimization problem is formulated in a slightly different way, which allows us to explicitly solve the problem using elementary optimization techniques.

3.3.4 Stability issues

The maximal stability region characterizes the set of arrival rates for which a static policy can keep all the queues of the system stable. Clearly for static policies, we have two independent PS queues, each with two classes of users. Therefore a system specified by an arrival rate vector $\boldsymbol{\lambda} = (\lambda_1^{\mathrm{n}}, \lambda_1^{\mathrm{f}}, \lambda_2^{\mathrm{n}}, \lambda_2^{\mathrm{f}}) \geq 0$ can be stabilized by a static policy with service rates $\boldsymbol{\phi} = (\phi_1, \phi_2)$ if and only if $\lambda_i^{\mathrm{n}} + \frac{\lambda_i^{\mathrm{f}}}{k} < \phi_i$ for all i, which is equivalent to the conditions

$$\begin{aligned}
& \frac{1}{\mu_0}\left(\lambda_1^{\mathrm{n}} + \frac{\lambda_1^{\mathrm{f}}}{k}\right) + \left(\frac{1}{\mu_1} - \frac{1}{\mu_0}\right)\left(\lambda_2^{\mathrm{n}} + \frac{\lambda_2^{\mathrm{f}}}{k}\right) < 1 \quad \text{and} \\
& \frac{1}{\mu_0}\left(\lambda_2^{\mathrm{n}} + \frac{\lambda_2^{\mathrm{f}}}{k}\right) + \left(\frac{1}{\mu_1} - \frac{1}{\mu_0}\right)\left(\lambda_1^{\mathrm{n}} + \frac{\lambda_1^{\mathrm{f}}}{k}\right) < 1.
\end{aligned}$$

This is the maximal stability region for the static policies and is shown in Figure 3.2. This is also the maximal stability region of the dynamic policies [86]. Note that the stability region is the same as the interior of the feasible rate region $\mathcal{C}$, when the axes are labeled as arrival rates.

3.3.5 Dynamic policies

In a dynamic policy, π, we choose an action $a^{\pi}(\boldsymbol{x})$ that depends on the state $\boldsymbol{x}$ of the system, and perhaps on other system parameters. The performance of these policies is then compared against the optimal static policy. We describe two such policies here.

Both Stations On policy

Both Stations On (BSO) is a dynamic policy that utilizes just the state information. We use the 'dd' mode when there are flows present in both the cells. If flows are present in just one cell, then either '0d' or 'd0'—whichever turns off the station without any flows—is utilized. More precisely, we choose action $a^{\mathrm{BSO}}(\boldsymbol{x})$ such that

$$a^{\mathrm{BSO}}(\boldsymbol{x}) = \begin{cases} \mathrm{dd}, & \text{if} \quad x_1^{\mathrm{n}} + x_1^{\mathrm{f}} \neq 0 \quad \text{and} \quad x_2^{\mathrm{n}} + x_2^{\mathrm{f}} \neq 0, \\ \mathrm{d0}, & \text{if} \quad x_1^{\mathrm{n}} + x_1^{\mathrm{f}} \neq 0 \quad \text{and} \quad x_2^{\mathrm{n}} + x_2^{\mathrm{f}} = 0, \\ \mathrm{0d}, & \text{if} \quad x_1^{\mathrm{n}} + x_1^{\mathrm{f}} = 0 \quad \text{and} \quad x_2^{\mathrm{n}} + x_2^{\mathrm{f}} \neq 0 \end{cases}$$

When only one class of users is present in both cells, this policy maximizes both the service rates and flow departure rates, and is even known to be stochastically optimal [95]. However, when two classes of users are present, although it maximizes the service rates, the flow departure rates are not necessarily maximized. This can be seen in a case when only near users are present in Station 1 and only far users are present in Station 2 and $\frac{\mu_0}{\mu_1} > 1 + k$. Under such conditions, the flow departure rate in the 'dd' mode $\mu_1(1+k) < \mu_0$, the flow departure rate in the 'd0' mode.

FPI policy

In the MDP-based approach, a baseline policy is improved by the method of *policy iteration* which follows from the theory of Markov Decision Processes [70]. We discussed a similar approach for a discrete-time system in Section 2.4.3. Here we present the continuous-time counterpart of the same technique.

In the policy iteration algorithm, the initial policy π is used to construct

a new policy π', for which the action, $a^{\pi'}(\boldsymbol{x})$, in state $\boldsymbol{x}$ is given by

$$a^{\pi'}(\boldsymbol{x}) = \arg\min_{a \in \mathcal{A}} \left\{ r_{\boldsymbol{x}}(a) - r(\pi) + \sum_{\boldsymbol{y} \neq \boldsymbol{x}} q_{\boldsymbol{x},\boldsymbol{y}}(a) \left(v_{\boldsymbol{y}}(\pi) - v_{\boldsymbol{x}}(\pi) \right) \right\}. \qquad (3.6)$$

The expression inside the braces gives the future cost rate of the system when the action a is chosen at the current decision instant and policy π is followed thereafter. Here $r_{\boldsymbol{x}}(a)$ is the instantaneous cost rate of state $\boldsymbol{x}$ with the newly chosen action a, $r(\pi)$ is the average cost rate under the initial policy π, $q_{\boldsymbol{x},\boldsymbol{y}}(a)$ is the transition rate from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ with the newly chosen action a, and $v_{\boldsymbol{x}}(\pi)$ is the relative value of state $\boldsymbol{x}$ under the initial policy π, which describes the expected cumulative difference in costs between a system starting from state $\boldsymbol{x}$ and a stationary system with the initial distribution equal to the equilibrium distribution under policy π.

We take the optimal static policy as our initial policy π and derive an improved dynamic policy π' (FPI) by applying the policy iteration algorithm once. The state is given by $\boldsymbol{x}$, and the action space consists of the three operation modes, $\mathcal{A} = \{0\mathrm{d}, \mathrm{d}0, \mathrm{dd}\}$. The instantaneous cost rate $r_{\boldsymbol{x}}(a)$ is equal to the total number of flows in the whole system, $|\boldsymbol{x}|$, and is thus independent of the chosen action a. As we have two parallel PS queues, the relative value of state of the system is the sum of the relative values of each PS queue.

To find the relative values of states of M/M/1-PS queue with two user-classes, we use the value function extrapolation technique using polynomial functions as described in [58]. This method allows the calculation of the exact relative values of states in a discriminatory processor sharing system. In Publication IV, we provide the details of the method used, where the coefficients of these polynomial value functions are determined by substituting them in the Howard equations. The FPI policy chooses an action $a^{\mathrm{FPI}}(\boldsymbol{x})$ in the following way:

$$a^{\mathrm{FPI}}(\boldsymbol{x}) = \begin{cases} 0\mathrm{d}, & \text{if} \quad t(\boldsymbol{x}) \leq \beta, \\ \mathrm{dd}, & \text{if} \quad \beta \leq t(\boldsymbol{x}) \leq \alpha, \\ \mathrm{d}0, & \text{if} \quad t(\boldsymbol{x}) \geq \alpha, \end{cases}$$

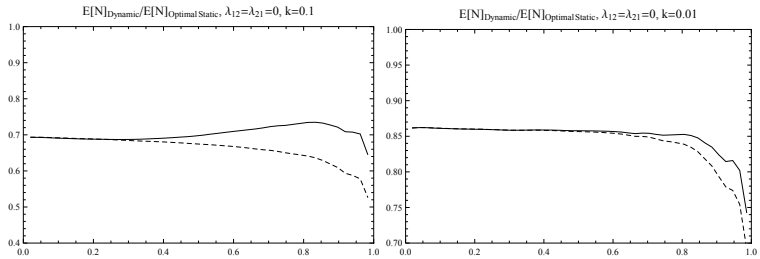


Figure 3.3. Asymmetric user arrivals with $k = 0.1$ (left) and $k = 0.01$ (right). The solid line represents the FPI policy and the dotted line represents the BSO policy. In the horizontal axis, the arrival rates are plotted normalized to the maximum possible arrival rate that keeps the system stable.

where,

$$t(\boldsymbol{x}) = \frac{(1+k)(x_1^{\mathrm{n}} + x_1^{\mathrm{f}}) - (x_1^{\mathrm{n}} + kx_1^{\mathrm{f}})\rho_1^*}{(1+k)(x_2^{\mathrm{n}} + x_2^{\mathrm{f}}) - (x_2^{\mathrm{n}} + kx_2^{\mathrm{f}})\rho_2^*},$$

$$\alpha = \frac{c_1}{c_0 - c_1}\frac{1-\rho_1^*}{1-\rho_2^*}\frac{(1+k)\phi_1^* - \lambda_1^{\mathrm{n}} - \lambda_1^{\mathrm{f}}}{(1+k)\phi_2^* - \lambda_2^{\mathrm{n}} - \lambda_2^{\mathrm{f}}},$$

$$\beta = \frac{c_0 - c_1}{c_1}\frac{1-\rho_1^*}{1-\rho_2^*}\frac{(1+k)\phi_1^* - \lambda_1^{\mathrm{n}} - \lambda_1^{\mathrm{f}}}{(1+k)\phi_2^* - \lambda_2^{\mathrm{n}} - \lambda_2^{\mathrm{f}}},$$

$$\rho_i^* = \frac{\lambda_i^{\mathrm{n}} + \frac{\lambda_i^{\mathrm{f}}}{k}}{\phi_i^*}.$$

It should be noted that the switching curves of the policy turn out to be simple linear functions of the state variables $\boldsymbol{x}$. This is because the relative values of states in this case are second-degree polynomials on the state of the system, and the difference of these functions are taken while determining the switching curves.

3.3.6 Numerical results

We now observe numerically the performance of the FPI policy compared to the optimal static policy, and then compare this improvement with the BSO policy. For the optimal static policy, the total average number of users is determined analytically, while for the two other policies we have conducted extensive simulations to estimate the corresponding mean values. The ratio between the total average number of users of the two dynamic policies and that of the optimal static policy is plotted in Figure 3.3 for two different values for the path loss parameter k. The arrival rates are set $\lambda_1^{\mathrm{f}} = \lambda_2^{\mathrm{n}} = 0$ and $\lambda_1^{\mathrm{n}} = \lambda_2^{\mathrm{f}} = \lambda$ with λ varying from 0 to the stability limit. The service rates are $\mu_0 = 5$ and $\mu_1 = 3$. Note that we have taken an extreme case with only the near users in the first station and only the far users in the second station. In such a case, FPI is as good as BSO at lower values of arrival rates. At higher arrival rates, BSO is better but

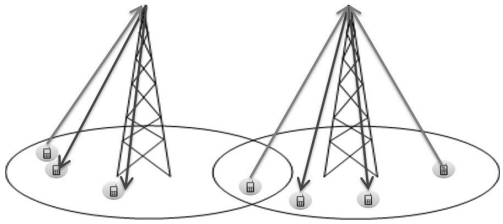


Figure 3.4. Interfering base stations with uplink and downlink users.

not by very much.

In a more general scenario with more balanced arrivals in all classes of both stations, we observe that BSO is again performing relatively better than FPI. This is true even for low values of the load. The results are presented in Publication IV.

3.4 Coordination in dynamic TDD-LTE systems

In Publication V and Publication VI, we consider two neighboring base stations operating in dynamic TDD mode serving uplink and downlink traffic as illustrated in Figure 3.4. Therefore, it is possible to dynamically change the number of subframes allocated to the uplink and the downlink data transmissions. We now have two queues in each station—an uplink queue and a downlink queue—making a total of four queues which need to be coordinated. Again we use flow-level models and PS queues to study the coordination problem.

3.4.1 Research problem

We study the problem of optimally coordinating the activities of two neighboring base stations operating in dynamic TDD configuration that interfere with each other. Each base station has two classes of users, viz., the uplink users and the downlink users with elastic traffic. In dynamic TDD configuration, a base station can dynamically allocate its resources either for the uplink or the downlink transmission. However, this decision of allocating transmission direction by one station affects the service of the users in the *other* cell. Based on the information about various system parameters and load conditions of the cells, the activities of the two base stations can be controlled. This can ensure that the operation of one does not adversely affect the other. The research problem we focus on here is the evaluation of the performance of various strategies, which are either

systematic techniques or heuristic policies, that aid in solving this coordination problem, as well as possible.

3.4.2 System model

We describe a model similar to the one described in Section 3.3.2 that is useful to formulate the problem in this setting. Consider two neighboring base stations, labeled 1 and 2 respectively, operating in the dynamic TDD configuration. Both stations, in their respective cells, serve the uplink (u) and the downlink (d) users with elastic flows. Thus, there are a total of four classes of flows, labeled by the pair (δ, i) where $\delta \in \{\mathrm{u}, \mathrm{d}\}$ is the service class of the flow while $i \in \{1, 2\}$ is the station the flow is attached to. In each class (δ, i), the flows arrive according to independent Poisson processes of rate λ_i^δ and have random sizes B_i^δ (bits) which are assumed to be independent and identically distributed with an arbitrary distribution. Furthermore, since each station can operate in either the 'u'-direction or the 'd'-direction, the *operating mode*[6] of the system, m, can take any value in $\{\mathrm{dd}, \mathrm{du}, \mathrm{ud}, \mathrm{uu}\}$[7], where the first letter of the mode denotes the operating direction of Station 1 while the second letter denotes the operating direction of Station 2. For example, when $m = \mathrm{ud}$, Station 1 is operating in the uplink mode while Station 2 is operating in the downlink mode.

Let r_i^m denote the service rate (in bits per second) provided by Station i operating in mode m to the flows in class (m_i, i), where $m_i \in \{\mathrm{u}, \mathrm{d}\}$ is the direction of the Station i flows in mode m, given by ith letter of m. We define the corresponding service completion rates μ_i^m as

$$\mu_i^m = \frac{r_i^m}{\mathbf{E}\left[B_i^{m_i}\right]} \tag{3.7}$$

with the following assumptions:

- Because of uplink/downlink *power asymmetry* the downlink power effectively destroys the uplink transmission on neighboring cell due to interference when operated in either the 'ud' or the 'du' modes, i.e.,

$$\mu_2^{\mathrm{du}} = \mu_1^{\mathrm{ud}} = 0.$$

- The intercell interference for downlink is higher in the 'dd' mode com-

[6] 'Operating mode' is frequently shortened to 'mode'.

[7] There can be an extra '00' mode if none of the queues are served. Since the system is non-idling, it is never used whenever there is at least one user in the system.

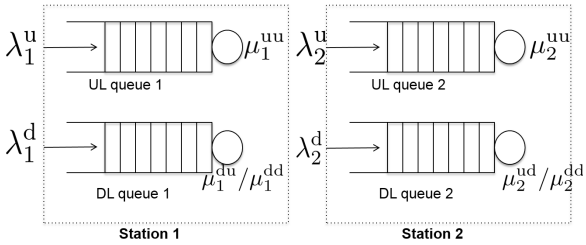


Figure 3.5. Interacting queues with uplink and downlink users. The service rate of each queue (μ_i^m) depends on the selected mode $m \in \{\mathrm{dd}, \mathrm{du}, \mathrm{ud}, \mathrm{uu}\}$.

pared to that in either the 'ud' or 'du' modes in both cells. Thus, the downlink service rates are higher in 'du' or 'ud' than in 'dd' i.e.,

$$\mu_1^{\mathrm{du}} > \mu_1^{\mathrm{dd}}, \quad \mu_2^{\mathrm{ud}} > \mu_2^{\mathrm{dd}}.$$

- We again define the *low interference* condition in the downlink as

$$\frac{\mu_1^{\mathrm{dd}}}{\mu_1^{\mathrm{du}}} + \frac{\mu_2^{\mathrm{dd}}}{\mu_2^{\mathrm{ud}}} \geq 1. \tag{3.8}$$

We also define a *symmetric* service case where the stations have identical service rates, i.e.,

$$\begin{aligned} \mu_1^{\mathrm{dd}} &= \mu_2^{\mathrm{dd}} = \mu^{\mathrm{dd}}, \\ \mu_1^{\mathrm{du}} &= \mu_2^{\mathrm{ud}} = \mu^{\mathrm{d}}, \\ \mu_1^{\mathrm{uu}} &= \mu_2^{\mathrm{uu}} = \mu^{\mathrm{uu}}. \end{aligned} \tag{3.9}$$

for which the low interference condition (3.8) becomes $2\mu^{\mathrm{dd}} \geq \mu^{\mathrm{d}}$.

We denote the instantaneous state of the system at time t by vector $\boldsymbol{X}(t) = (X_1^{\mathrm{d}}(t), X_1^{\mathrm{u}}(t), X_2^{\mathrm{d}}(t), X_2^{\mathrm{u}}(t))$ where $X_i^{\delta}(t)$ is the number of flows in class (δ, i). Within each class the flows are fairly served, e.g., in a round-robin manner, which attempts to provide an equal number of timeslots to all flows. With the additional assumption of time-scale separation between the flow dynamics and timeslot-level channel dynamics, the users are served according to the PS queueing discipline [30]. Thus, we have four interacting queues (see Figure 3.5) with independent but random arrivals and coupled service rates, which depend on different operating modes.

3.4.3 Optimal coordination problem

The optimal coordination problem is to find the ways to operate the different modes of the stations, based on the information about the arrival

rates, the service rates and the queue lengths, so that the average flow-level delay is minimized. We again start with the static coordination policies which choose constant probability p^m to operate mode m in every timeslot. Then, in Section 3.4.5, we discuss the *dynamic policies*.

Let $\boldsymbol{p} = (p^{\mathrm{dd}}, p^{\mathrm{du}}, p^{\mathrm{ud}}, p^{\mathrm{uu}})$ be a vector whose components p^m represent the probability that the system is run in mode m in any timeslot. These components can also be interpreted as the long-term fraction the system is run in the respective mode. Thus we have

$$0 \leq \boldsymbol{p} \leq 1 \quad \text{and} \tag{3.10}$$

$$|\boldsymbol{p}| \leq 1. \tag{3.11}$$

Again, the strict inequality in (3.11) can occur if the system is idling. The equality holds for a non-idling system.

Since $\boldsymbol{p}$ is constant, the four PS queues in the two stations are no longer coupled, and thus, we get four independent PS queues. The service rate ϕ_i^δ of class (δ, i), is then given by

$$\begin{aligned} \phi_1^{\mathrm{d}} &= p^{\mathrm{dd}}\mu_1^{\mathrm{dd}} + p^{\mathrm{du}}\mu_1^{\mathrm{du}}, \quad \phi_1^{\mathrm{u}} = p^{\mathrm{uu}}\mu_1^{\mathrm{uu}}, \\ \phi_2^{\mathrm{d}} &= p^{\mathrm{dd}}\mu_2^{\mathrm{dd}} + p^{\mathrm{ud}}\mu_2^{\mathrm{ud}}, \quad \phi_2^{\mathrm{u}} = p^{\mathrm{uu}}\mu_2^{\mathrm{uu}}, \end{aligned} \tag{3.12}$$

and these service rates are collected together as the component of the vector $\boldsymbol{\phi} = (\phi_1^{\mathrm{d}}, \phi_1^{\mathrm{u}}, \phi_2^{\mathrm{d}}, \phi_2^{\mathrm{u}})$. Note that these relations are valid only if r_i^m is constant for class (m_i, i) when the stations are operated exclusively in mode m. In Section 3.4.7 we discuss a physical model in which the service rates are based on the locations of the users in the cell for all modes.

Therefore, the total average queue length is calculated as

$$\begin{aligned} \mathbf{E}[|\boldsymbol{X}|] = \; & \frac{\lambda_1^{\mathrm{u}}}{p^{\mathrm{uu}}\mu_1^{\mathrm{uu}} - \lambda_1^{\mathrm{u}}} + \frac{\lambda_1^{\mathrm{d}}}{p^{\mathrm{du}}\mu_1^{\mathrm{du}} + p^{\mathrm{dd}}\mu_1^{\mathrm{dd}} - \lambda_1^{\mathrm{d}}} \\ & + \frac{\lambda_2^{\mathrm{u}}}{p^{\mathrm{uu}}\mu_2^{\mathrm{uu}} - \lambda_2^{\mathrm{u}}} + \frac{\lambda_2^{\mathrm{d}}}{p^{\mathrm{ud}}\mu_2^{\mathrm{ud}} + p^{\mathrm{dd}}\mu_2^{\mathrm{dd}} - \lambda_2^{\mathrm{d}}}. \end{aligned} \tag{3.13}$$

The set of all service rates $\boldsymbol{\phi}$ that satisfy (3.10), (3.11) and (3.12) is called the *capacity region* and is denoted by $\mathcal{C}$. In this case, the service rate vector $\boldsymbol{\phi}$ satisfies the following set of inequalities as mentioned in Publication VI:

$$\begin{aligned} a_1\phi_1^{\mathrm{d}} + b_1\phi_2^{\mathrm{d}} + c_1\phi_1^{\mathrm{u}} \leq 1, \quad a_1\phi_1^{\mathrm{d}} + b_1\phi_2^{\mathrm{d}} + c_2\phi_2^{\mathrm{u}} \leq 1, \\ b_2\phi_1^{\mathrm{d}} + a_2\phi_2^{\mathrm{d}} + c_1\phi_1^{\mathrm{u}} \leq 1, \quad b_2\phi_1^{\mathrm{d}} + a_2\phi_2^{\mathrm{d}} + c_2\phi_2^{\mathrm{u}} \leq 1, \end{aligned} \tag{3.14}$$

where

$$\begin{aligned} a_1 &= \frac{1}{\mu_1^{\mathrm{du}}}, \quad b_1 = \frac{\mu_1^{\mathrm{du}} - \mu_1^{\mathrm{dd}}}{\mu_1^{\mathrm{du}}\mu_2^{\mathrm{dd}}}, \quad c_1 = \frac{1}{\mu_1^{\mathrm{uu}}}, \\ a_2 &= \frac{1}{\mu_2^{\mathrm{ud}}}, \quad b_2 = \frac{\mu_2^{\mathrm{ud}} - \mu_2^{\mathrm{dd}}}{\mu_2^{\mathrm{ud}}\mu_1^{\mathrm{dd}}}, \quad c_2 = \frac{1}{\mu_2^{\mathrm{uu}}}, \end{aligned}$$

which is a four-dimensional polytope in the service rate space.

The *optimal static policy* is obtained by properly selecting the components of $\boldsymbol{p}$ that minimizes the average number of flows calculated in (3.13), and by Little's result the average flow-level delay, as well, i.e.,

$$\begin{aligned}\boldsymbol{p}^* = \arg\min_{\boldsymbol{p}} \Bigg\{ & \frac{\lambda_1^{\mathrm{u}}}{p^{\mathrm{uu}}\mu_1^{\mathrm{uu}} - \lambda_1^{\mathrm{u}}} + \frac{\lambda_1^{\mathrm{d}}}{p^{\mathrm{du}}\mu_1^{\mathrm{du}} + p^{\mathrm{dd}}\mu_1^{\mathrm{dd}} - \lambda_1^{\mathrm{d}}} \\ & + \frac{\lambda_2^{\mathrm{u}}}{p^{\mathrm{uu}}\mu_2^{\mathrm{uu}} - \lambda_2^{\mathrm{u}}} + \frac{\lambda_2^{\mathrm{d}}}{p^{\mathrm{ud}}\mu_2^{\mathrm{ud}} + p^{\mathrm{dd}}\mu_2^{\mathrm{dd}} - \lambda_2^{\mathrm{d}}} \Bigg\}, \\ \text{s.t.} \quad & 0 \le \boldsymbol{p} \le 1, \qquad |\boldsymbol{p}| \le 1, \\ & p^{\mathrm{uu}}\mu_1^{\mathrm{uu}} - \lambda_1^{\mathrm{u}} > 0, \quad p^{\mathrm{du}}\mu_1^{\mathrm{du}} + p^{\mathrm{dd}}\mu_1^{\mathrm{dd}} - \lambda_1^{\mathrm{d}} > 0, \\ & p^{\mathrm{uu}}\mu_2^{\mathrm{uu}} - \lambda_2^{\mathrm{u}} > 0, \quad p^{\mathrm{ud}}\mu_2^{\mathrm{ud}} + p^{\mathrm{dd}}\mu_2^{\mathrm{dd}} - \lambda_2^{\mathrm{d}} > 0.\end{aligned} \tag{3.15}$$

We can observe that (3.15) is a convex optimization problem whose solution for a general case has to be calculated numerically. However, for a few symmetric cases it is possible to determine the explicit expression of $\boldsymbol{p}^*$, as has been shown in Publication V. From $\boldsymbol{p}^*$, we can determine the optimal static service rates $\boldsymbol{\phi}^*$ using (3.12).

3.4.4 Stability issues

We now discuss the nature of the maximal stability region for the static policies. The maximal stability region characterizes the set of arrival rates for which a static policy can keep all the queues of the system stable. In Publication VI, (Theorem 1 in §4), we prove the following characterization the maximal stability region for a static policy:
For $\lambda_1^{\mathrm{d}}/\mu_1^{\mathrm{dd}} > \lambda_2^{\mathrm{d}}/\mu_2^{\mathrm{dd}}$*, there exists a stable static policy if and only if*

$$\max\left\{\frac{\lambda_1^{\mathrm{u}}}{\mu_1^{\mathrm{uu}}}, \frac{\lambda_2^{\mathrm{u}}}{\mu_2^{\mathrm{uu}}}\right\} + \frac{\lambda_2^{\mathrm{d}}}{\mu_2^{\mathrm{dd}}} + \frac{1}{\mu_1^{\mathrm{du}}}\left(\lambda_1^{\mathrm{d}} - \mu_1^{\mathrm{dd}}\frac{\lambda_2^{\mathrm{d}}}{\mu_2^{\mathrm{dd}}}\right) < 1. \tag{3.16}$$

If $\lambda_1^{\mathrm{d}}/\mu_1^{\mathrm{dd}} \le \lambda_2^{\mathrm{d}}/\mu_2^{\mathrm{dd}}$*, there exists a stable static policy if and only if*

$$\max\left\{\frac{\lambda_1^{\mathrm{u}}}{\mu_1^{\mathrm{uu}}}, \frac{\lambda_2^{\mathrm{u}}}{\mu_2^{\mathrm{uu}}}\right\} + \frac{\lambda_1^{\mathrm{d}}}{\mu_1^{\mathrm{dd}}} + \frac{1}{\mu_2^{\mathrm{ud}}}\left(\lambda_2^{\mathrm{d}} - \mu_2^{\mathrm{dd}}\frac{\lambda_1^{\mathrm{d}}}{\mu_1^{\mathrm{dd}}}\right) < 1. \tag{3.17}$$

Intuitively, the first term of (3.16) implies that for stability, as the two uplink arrivals are served only by the 'uu' mode, the 'uu' mode should have enough time to serve the larger of the two uplink arrival rates. Moreover, since $\lambda_1^{\mathrm{d}}/\mu_1^{\mathrm{dd}} > \lambda_2^{\mathrm{d}}/\mu_2^{\mathrm{dd}}$, the second term implies that the 'dd' mode should have enough time to serve the downlink arrivals in Station 2. Finally, since this 'dd' mode also serves the downlink arrivals in Station 1, after it is used long enough to clear the downlink arrivals in Station 2, the

remaining fraction of time can then be allocated to the 'du' mode to serve only the downlink arrivals in Station 1 at a higher rate. Clearly, all these fractions of time should add up to 1 or less for a feasible allocation. This is also the maximal stability region even for the dynamic policies [86].

Note that when $\lambda_1^{\mathrm{d}}/\mu_1^{\mathrm{dd}} > \lambda_2^{\mathrm{d}}/\mu_2^{\mathrm{dd}}$, the first term in the left hand side of (3.16) is defined as the total uplink load, ρ^{u}, while the remaining terms are together defined as the total downlink load ρ^{d}. The loads are analogously defined when $\lambda_1^{\mathrm{d}}/\mu_1^{\mathrm{dd}} \leq \lambda_2^{\mathrm{d}}/\mu_2^{\mathrm{dd}}$ using (3.17).

Finally, we again observe that the stability region (3.16) or (3.17) is the interior of the rate region given by (3.14), where the service rates ϕ_i^{δ} are replaced by the arrival rates λ_i^{δ}.

3.4.5 Dynamic policies

In a dynamic policy π, we choose an action $a^{\pi}(\boldsymbol{x})$ that depends on the state $\boldsymbol{x}$ of the system, and perhaps on other parameters. We classify the dynamic policies into three broad categories—the policies based on priority, the policies based on systematic techniques, and the policies based on well-performing heuristics. The priority based policies, HU and HD, that are inspired by the BSO policy, defined in Section 3.3.5, but adapted to dynamic TDD, are stochastically optimal in some cases. Techniques like MDP-based policy iteration algorithm and Balanced Fairness come from well-established principles and provide systematic ways to optimize the resource allocation. Among the heuristic policies, Max-Weight is well known for being maximally stable and even asymptotically optimal in discrete-time queues, and the Robust Policy attempts to equalize the two queues operating in two directions as much as possible, without using any information about the traffic parameters. This classification is also presented in Table 3.1.

Uplink priority policy (HU)

In HU, the uplink queues are served with the highest priority. Only when both the uplink queues are empty, either the 'dd' mode or one of the 'du' or 'ud' modes is used depending on the presence of the flows in both or just

Table 3.1. Classification of the dynamic policies

Classification of policies	**Name of policies**
Priority policies	Uplink priority policy (HU), Downlink priority policy (HD)
Policies based on systematic techniques	Markov Decision Processes based policy (MDP), Balanced Fair policy (BF)
Heuristic policies	Max-Weight policy (MW), Robust Policy (RP)

one of the downlink queues, respectively, i.e.,

$$a^{\mathrm{HU}}(\boldsymbol{x}) = \begin{cases} \mathrm{uu}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} > 0, \\ \mathrm{dd}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} = 0, x_1^{\mathrm{d}} x_2^{\mathrm{d}} > 0, \\ \mathrm{du}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} = 0, x_1^{\mathrm{d}} x_2^{\mathrm{d}} = 0, x_1^{\mathrm{d}} > 0, \\ \mathrm{ud}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} = 0, x_1^{\mathrm{d}} x_2^{\mathrm{d}} = 0, x_2^{\mathrm{d}} > 0. \end{cases}$$

Clearly, it is a good heuristic when the uplink flow service rates $\mu_1^{\mathrm{uu}}, \mu_2^{\mathrm{uu}}$ are high compared to the downlink flow service rates, μ_1^{du}, μ_2^{ud}, $\mu_1^{\mathrm{dd}}, \mu_2^{\mathrm{dd}}$, and the traffic load is sufficiently small. Indeed, we will later observe that HU is even stochastically optimal under some conditions with large enough μ_1^{uu} and μ_2^{uu}. However the absolute priority given to the uplink may render the downlink queues unstable at high loads.

Downlink priority policy (HD)

In HD, the downlink is given the highest priority, and the uplink ('uu') mode is chosen only when both the downlink queues are empty. The 'dd' mode is used when both downlink queues have flows, while one of 'du' or 'ud' is used only if one of the downlinks has flows. Thus,

$$a^{\mathrm{HD}}(\boldsymbol{x}) = \begin{cases} \mathrm{dd}, & \text{if } x_1^{\mathrm{d}} x_2^{\mathrm{d}} > 0, \\ \mathrm{du}, & \text{if } x_1^{\mathrm{d}} x_2^{\mathrm{d}} = 0, x_1^{\mathrm{d}} > 0, \\ \mathrm{ud}, & \text{if } x_1^{\mathrm{d}} x_2^{\mathrm{d}} = 0, x_2^{\mathrm{d}} > 0, \\ \mathrm{uu}, & \text{if } x_1^{\mathrm{d}} + x_2^{\mathrm{d}} = 0. \end{cases}$$

It is reasonable to assume that this policy will give good performance when the downlink flow completion rates, μ_1^{du}, μ_2^{ud}, $\mu_1^{\mathrm{dd}}, \mu_2^{\mathrm{dd}}$, are high compared to the uplink flow completion rate, μ_1^{uu} and μ_2^{uu}. It turns out to be stochastically optimal in some cases where the downlink service rates are

high enough, but may suffer from the instability of the downlink queues at high loads.

Max-Weight policy (MW)

In this policy, the mode that has the *maximum weight* for a state is chosen. The weight is defined as the sum of the product of the service completion rate of the queues in the chosen mode and the queue lengths of the same queues. Thus, in the MW policy, the action chosen in state $\boldsymbol{x}$ is defined by

$$a^{\mathrm{MW}}(\boldsymbol{x}) = \underset{\alpha\in\{\mathrm{uu,du,ud,dd}\}}{\arg\max} \; w^{\alpha}(\boldsymbol{x}),$$

where,

$$w^{\mathrm{dd}}(\boldsymbol{x}) = \mu_1^{\mathrm{dd}} x_1^{\mathrm{d}} + \mu_2^{\mathrm{dd}} x_2^{\mathrm{d}}, \quad w^{\mathrm{du}}(\boldsymbol{x}) = \mu_1^{\mathrm{du}} x_1^{\mathrm{d}},$$
$$w^{\mathrm{uu}}(\boldsymbol{x}) = \mu_1^{\mathrm{uu}} x_1^{\mathrm{u}} + \mu_2^{\mathrm{uu}} x_2^{\mathrm{u}}, \quad w^{\mathrm{ud}}(\boldsymbol{x}) = \mu_2^{\mathrm{ud}} x_2^{\mathrm{d}}.$$

Robust Policy (RP)

The robust policy we propose here relies solely on the information about the state of the system and is not purely priority based. It attempts to equalize the total queue lengths at the uplink and downlink as much as possible by choosing appropriate modes defined as

$$a^{\mathrm{RP}}(\boldsymbol{x}) = \begin{cases} \mathrm{uu}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} > x_1^{\mathrm{d}} + x_2^{\mathrm{d}}, \\ \mathrm{dd}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} < x_1^{\mathrm{d}} + x_2^{\mathrm{d}}, x_1^{\mathrm{d}} > 0, x_2^{\mathrm{d}} > 0 \\ \mathrm{du}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} < x_1^{\mathrm{d}} + x_2^{\mathrm{d}}, x_1^{\mathrm{d}} > 0, x_2^{\mathrm{d}} = 0, \\ \mathrm{ud}, & \text{if } x_1^{\mathrm{u}} + x_2^{\mathrm{u}} < x_1^{\mathrm{d}} + x_2^{\mathrm{d}}, x_1^{\mathrm{d}} = 0, x_2^{\mathrm{d}} > 0, \end{cases}$$

with an additional tie-breaking rule if $x_1^{\mathrm{u}} + x_2^{\mathrm{u}} = x_1^{\mathrm{d}} + x_2^{\mathrm{d}}$. This rule chooses the uplink mode ('uu') or an *appropriate* downlink mode ('dd', 'ud' or 'du'), depending on the queue lengths at different downlink queues, with equal probabilities.

MDP-based policies (FPI and MDP)

In the MDP-based approach, a baseline policy is improved by the method of *policy iteration* which follows from the theory of Markov Decision Processes [70], and has been discussed in Section 3.3.5. We note that, to apply this approach, we have to assume that the service times in each mode are exponentially distributed.

We take the optimal static policy as our initial policy π and derive an improved dynamic policy π' (FPI) by applying the policy iteration algorithm once. The state is given by $\boldsymbol{x}$, and the action space consists of the four

operation modes, $\mathcal{A} = \{\mathrm{uu}, \mathrm{ud}, \mathrm{du}, \mathrm{dd}\}$. The instantaneous cost rate $r_{\boldsymbol{x}}(a)$ is equal to the total number of flows in the whole system, $|\boldsymbol{x}|$. As we have parallel PS queues, the relative value of state of the system is the sum of the relative values of each PS queue. The relative value of state of an M/M/1-PS queue with service rate μ and arrival rate λ when there are n customers present is (see, e.g., [48])

$$v(n) = v(0) + \frac{n(n+1)}{2(\mu - \lambda)}.$$

The chosen action only affects the future departures from the system and has no effect on the average cost rate $r(\pi)$, the instantaneous cost rate $r_{\boldsymbol{x}}(a)$ and the future cost rate *only due to the arrivals* in the optimality equation (3.6). Thus, in each state $\boldsymbol{x}$ the action $a^{\mathrm{FPI}}(\boldsymbol{x})$ that minimizes the future cost rate related to the departure of a flow from the state is chosen, i.e.,

$$a^{\mathrm{FPI}}(\boldsymbol{x}) = \underset{\alpha \in \{\mathrm{dd},\mathrm{du},\mathrm{ud},\mathrm{uu}\}}{\arg\min} \; w^{\alpha}(\boldsymbol{x}),$$

where

$$w^{\mathrm{dd}}(\boldsymbol{x}) = -\frac{\mu_1^{\mathrm{dd}} x_1^{\mathrm{d}}}{\phi_1^{*\mathrm{d}} - \lambda_1^{\mathrm{d}}} - \frac{\mu_2^{\mathrm{dd}} x_2^{\mathrm{d}}}{\phi_2^{*\mathrm{d}} - \lambda_2^{\mathrm{d}}}, \quad w^{\mathrm{du}}(\boldsymbol{x}) = -\frac{\mu_1^{\mathrm{du}} x_1^{\mathrm{d}}}{\phi_1^{*\mathrm{d}} - \lambda_1^{\mathrm{d}}},$$
$$w^{\mathrm{uu}}(\boldsymbol{x}) = -\frac{\mu_1^{\mathrm{uu}} x_1^{\mathrm{u}}}{\phi_1^{*\mathrm{u}} - \lambda_1^{\mathrm{u}}} - \frac{\mu_2^{\mathrm{uu}} x_2^{\mathrm{u}}}{\phi_2^{*\mathrm{u}} - \lambda_2^{\mathrm{u}}}, \quad w^{\mathrm{ud}}(\boldsymbol{x}) = -\frac{\mu_2^{\mathrm{ud}} x_2^{\mathrm{d}}}{\phi_2^{*\mathrm{d}} - \lambda_2^{\mathrm{d}}},$$

and the components of $\boldsymbol{\phi}^* = (\phi_1^{*\mathrm{u}}, \phi_1^{*\mathrm{d}}, \phi_2^{*\mathrm{u}}, \phi_2^{*\mathrm{d}})$ are the optimal static service rates of the different queues.

Note that, initially, when the optimal static policy is used, all the queues are independent and the relative values of the states have simple expressions, which are then added together. The process obtained after the application of the policy iteration algorithm for the first time couples the system, and thus, the four queues are no longer independent. Consequently, the relative values of the states for this coupled system no longer have simple explicit expressions after the first iterated policy is applied. However, the policy iteration algorithm can be continued numerically, for a *truncated system*, until convergence, when the policy does not change with each successive iteration. Naturally, these numerical evaluations are prone to inaccuracies, as those for the truncated systems consistently underestimate the average queue lengths. These inaccuracies are more prominent at high loads than at the low loads. This policy, obtained by running the policy iteration algorithm to its convergence, is labeled as 'MDP'.

Optimal balanced dynamic policy (BF)

We define the optimal balanced dynamic policy based on the notion of *Balanced Fairness* [23, 24, 26, 27, 68]. A coordination policy is said to be *balanced* if, for any classes (δ, i) and (δ', i') and any state x such that $\boldsymbol{x}_i^{\delta}, \boldsymbol{x}_{i'}^{\delta'} > 0$,

$$\frac{\phi_i^{\delta}(\boldsymbol{x} - \boldsymbol{e}_{i'}^{\delta'})}{\phi_i^{\delta}(\boldsymbol{x})} = \frac{\phi_{i'}^{\delta'}(\boldsymbol{x} - \boldsymbol{e}_i^{\delta})}{\phi_{i'}^{\delta'}(\boldsymbol{x})},$$

where $\boldsymbol{e}_i^{\delta}$ is the state with only one flow in (δ, i) class and no flows in any other class. It is shown in [23] that a coordination policy is balanced if and only if there is a so-called *balance function* $\Phi(\boldsymbol{x})$, which is a bounded and non-negative function satisfying, for any class (δ, i) and any state $\boldsymbol{x}$ such that $x_i^{\delta} > 0$,

$$\phi_i^{\delta}(\boldsymbol{x}) = \frac{\Phi(\boldsymbol{x} - \boldsymbol{e}_i^{\delta})}{\Phi(\boldsymbol{x})}. \tag{3.18}$$

From (3.18), we get the so-called *detailed balance equations*

$$\lambda_i^{\delta} \pi(\boldsymbol{x} - \boldsymbol{e}_i^{\delta}) = \phi_i^{\delta}(\boldsymbol{x}) \pi(\boldsymbol{x}), \tag{3.19}$$

where we have defined

$$\pi(x) = \frac{1}{G} (\lambda_1^{\mathrm{d}})^{x_1^{\mathrm{d}}} (\lambda_1^{\mathrm{u}})^{x_1^{\mathrm{u}}} (\lambda_2^{\mathrm{d}})^{x_2^{\mathrm{d}}} (\lambda_2^{\mathrm{u}})^{x_2^{\mathrm{u}}} \Phi(\boldsymbol{x}),$$

and normalization constant G is chosen so that $\pi(\boldsymbol{x})$ is a proper distribution in the state space. As an immediate consequence of (3.19), we observe that $\pi(\boldsymbol{x})$ is the equilibrium distribution of the system state if the service times are exponential. However, due to the *insensitivity* property of balanced allocations, the result is even valid for any service time distribution [23, 24].

The *balanced fair* (BF) coordination policy is defined by (3.18), where the corresponding balance function $\Phi(\boldsymbol{x})$ is given as the solution of the following recursion: $\Phi(0,0,0,0) = 1$, and for any state $\boldsymbol{x} \neq (0,0,0,0)$,

$$\Phi(\boldsymbol{x}) = \max \left\{ \alpha > 0 \middle| \left(\frac{\Phi(\boldsymbol{x} - \boldsymbol{e}_1^{\mathrm{d}})}{\alpha}, \frac{\Phi(\boldsymbol{x} - \boldsymbol{e}_1^{\mathrm{u}})}{\alpha}, \frac{\Phi(\boldsymbol{x} - \boldsymbol{e}_2^{\mathrm{d}})}{\alpha}, \frac{\Phi(\boldsymbol{x} - \boldsymbol{e}_2^{\mathrm{u}})}{\alpha} \right) \in \mathcal{C} \right\},$$

where $\Phi(\boldsymbol{x}) = 0$ if $\boldsymbol{x}$ does not belong to the state space. The BF coordination policy is the optimal balanced coordination policy in the sense that all the constraints are taken into account as tight as possible.

In Theorem 2 of Publication VI, we show that the four-dimensional recursion needed for BF calculation for our model can be reduced to a simpler 2-dimensional recursion, and calculate the BF balance function $\Phi(\boldsymbol{x})$ for any state $\boldsymbol{x}$ recursively as

$$\Phi(\boldsymbol{x}) = \binom{x_1^{\mathrm{d}} + x_1^{\mathrm{u}} + x_2^{\mathrm{d}} + x_2^{\mathrm{u}}}{x_1^{\mathrm{u}} + x_2^{\mathrm{u}}} \left(\frac{1}{\mu_1^{\mathrm{uu}}} \right)^{x_1^{\mathrm{u}}} \left(\frac{1}{\mu_2^{\mathrm{uu}}} \right)^{x_2^{\mathrm{u}}} \Phi(x_1^{\mathrm{d}}, x_2^{\mathrm{d}}),$$

where $\Phi(0,0) = 1$, $\Phi(u,v) = 0$ if $u < 0$ or $v < 0$, and for any state $(u,0,v,0) \neq (0,0,0,0)$,

$$\Phi(u,v) = \max\left\{ \frac{1}{\mu_1^{\mathrm{du}}}\Phi(u-1,y) + \frac{\mu_1^{\mathrm{du}} - \mu_1^{\mathrm{dd}}}{\mu_1^{\mathrm{du}}\mu_2^{\mathrm{dd}}}\Phi(u,v-1), \right.$$
$$\left. \frac{\mu_2^{\mathrm{ud}} - \mu_2^{\mathrm{dd}}}{\mu_2^{\mathrm{ud}}\mu_1^{\mathrm{dd}}}\Phi(u-1,v) + \frac{1}{\mu_2^{\mathrm{ud}}}\Phi(u,v-1) \right\}.$$

which can then be used to determine the balanced fair allocation in (3.18). For a symmetric system (3.9), we have explicit expressions for the balance functions given by

$$\Phi(\boldsymbol{x}) = \Phi(x_1^{\mathrm{d}}, x_1^{\mathrm{u}}, x_2^{\mathrm{d}}, x_2^{\mathrm{u}}) = \binom{x_1^{\mathrm{d}} + x_1^{\mathrm{u}} + x_2^{\mathrm{d}} + x_2^{\mathrm{u}}}{x_1^{\mathrm{u}} + x_2^{\mathrm{u}}} \left(\frac{1}{\mu_{\mathrm{uu}}}\right)^{(x_1^{\mathrm{u}} + x_2^{\mathrm{u}})} \Phi(x_1^{\mathrm{d}}, x_2^{\mathrm{d}})$$

where $\Phi(0,0) = 1$, for any state $(u,0,v,0) \neq (0,0,0,0)$ such that $u > v$,

$$\Phi(u,v) = \Phi(v,u)$$
$$= \sum_{i=0}^{v} \binom{u-1+v-1}{v-i} \left(\frac{1}{\mu_{\mathrm{d}}}\right)^{u} \left(\frac{1}{\mu_{\mathrm{dd}}} - \frac{1}{\mu_{\mathrm{d}}}\right)^{v-i} \left(\frac{1}{\mu_{\mathrm{dd}}}\right)^{i} \frac{u-v+i}{u}.$$

3.4.6 Stochastic optimality results

When the service times are symmetric (3.9) and exponentially distributed, under certain special conditions of the arrival rates of the flows, we can prove two results related to *stochastically optimal policies*. These can be seen as extensions of a similar result for single class user presented in [95].

A policy π^* is said to be stochastically optimal if for any $s, \boldsymbol{x}$ and t,

$$\pi^* = \arg\min_{\pi \in \Pi} \mathbf{P}\left\{ \left| \boldsymbol{X}^{\pi}(t) \right| > s \mid \boldsymbol{X}^{\pi}(0) = \boldsymbol{x} \right\}$$

where $\boldsymbol{X}^{\pi}(t)$ is the state of the system under policy π at time t. This means that, starting from any state $\boldsymbol{x}$, the total queue length process has the minimum tail probability when the stochastically optimal policy is used. For a general case, the stochastically optimal policy in our system is not known. We state the results related to stochastic optimality for some special cases in Theorem 3 and Theorem 4 of Publication VI, which are stated below as well.

When there are no uplink arrivals in one of the stations ($\lambda_1^{\mathrm{u}} > 0$ and $\lambda_2^{\mathrm{u}} = 0$), and the downlink service rates are sufficiently high ($\mu^{\mathrm{uu}} \geq 2\mu^{\mathrm{dd}} \geq \mu^{\mathrm{d}}$), then the HU policy is stochastically optimal.

The HU policy is stochastically optimal if

$$\lambda_1^{\mathrm{u}} > 0, \quad \lambda_2^{\mathrm{u}} = 0 \quad \textit{and} \quad \mu^{\mathrm{uu}} \geq 2\mu^{\mathrm{dd}} \geq \mu^{\mathrm{d}}.$$

On the other hand, when there are no downlink streams in one of the stations ($\lambda_1^{\mathrm{d}} > 0$, $\lambda_2^{\mathrm{d}} = 0$) and the downlink service completion rates are sufficiently high ($\mu^{\mathrm{d}} \geq 2\mu^{\mathrm{uu}}$), then the HD policy, that prioritizes the downlink, is stochastically optimal.

The HD policy is stochastically optimal if

$$\lambda_1^{\mathrm{d}} > 0, \quad \lambda_2^{\mathrm{d}} = 0 \quad \textit{and} \quad \mu^{\mathrm{d}} \geq 2\mu^{\mathrm{uu}}.$$

The proofs are based on induction and are detailed in Publication VI. They are obtained by uniformizing the Markov process and then using dynamic programming techniques to arrive at the result.

3.4.7 Physical model

We now describe a physical model in a wireless system that serves as a motivation for studying the abstract problem presented in Section 3.4.2. The flows are all assumed to arrive for service according to independent Poisson processes inside the cell. Thus, for a given number of flows in a cell, the users are uniformly distributed in the cell.

When a dynamic policy is used, the operating modes can change only at the arrival or departure of the flows. Thus, the operating modes change at a much slower rate than the time slots are scheduled. Due to this time-scale separation, each Station i has a well defined mean transmission rate $c_i^m(r)$ for a user located at a distance r from the station in mode m. When a user is at random distance R_i from Station i, the service time S_i^m of its flow, assuming that it is the only user present and the system operates exclusively in mode m, is given by

$$S_i^m = \frac{B_i^{m_i}}{c_i^m(R_i)}.$$

Recall that $m_i \in \{\mathrm{u}, \mathrm{d}\}$ is the service direction of Station i in mode m. If any station is not transmitting in a particular mode ($c_i^m(R_i) = 0$) then the service times are defined to be infinite, implying that the corresponding service rates are 0. Assuming that the size of the flow $B_i^{m_i}$ and the location R_i are independent of each other we get

$$\mathbf{E}[S_i^m] = \mathbf{E}[B_i^{m_i}]\, \mathbf{E}\left[\frac{1}{c_i^m(R_i)}\right]. \tag{3.20}$$

Thus the service completion rate provided by Station i operating in mode m under a dynamic policy is given by

$$\mu_i^m = \frac{1}{\mathbf{E}[S_i^m]} = \frac{\mathbf{E}[\frac{1}{c_i^m(R_i)}]^{-1}}{\mathbf{E}[B_i^{m_i}]},$$

which is equivalent to the definition (3.7) when the constant transmission rate r_i^m is replaced by the harmonic mean of the transmission rates with respect to the spatial distribution of the flows.

However, a static policy $\boldsymbol{p} = (p^{\mathrm{dd}}, p^{\mathrm{du}}, p^{\mathrm{ud}}, p^{\mathrm{uu}})$ chooses probabilistically a mode at every time slot that leads to a very fast switching of the modes compared with the time scale at which the flow-level events occur. This implies that a user at any location receives a time-averaged bit rate that depends on the distribution of the different modes in a time slot. The mean service rate of the whole class is then determined by taking into account the *spatial* distribution of the users. The mean service times of Station i under static policy $\boldsymbol{p} = (p^{\mathrm{dd}}, p^{\mathrm{du}}, p^{\mathrm{ud}}, p^{\mathrm{uu}})$ are given by (cf. [73, 74])

$$\frac{1}{\phi_i^{\mathrm{u}}} = \mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{p^{\mathrm{uu}} c_i^{\mathrm{uu}}(R_i)}\right] = \frac{1}{p^{\mathrm{uu}}}\mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{c_i^{\mathrm{uu}}(R_i)}\right], \tag{3.21a}$$

$$\frac{1}{\phi_1^{\mathrm{d}}} = \mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{p^{\mathrm{dd}} c_1^{\mathrm{dd}}(R_1) + p^{\mathrm{du}} c_1^{\mathrm{du}}(R_1)}\right], \tag{3.21b}$$

$$\frac{1}{\phi_2^{\mathrm{d}}} = \mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{p^{\mathrm{dd}} c_2^{\mathrm{dd}}(R_2) + p^{\mathrm{ud}} c_2^{\mathrm{ud}}(R_2)}\right]. \tag{3.21c}$$

We see that the uplink service rates ϕ_i^{u} in (3.12) are exactly the same as in (3.21a) since there is just one mode that serves the uplink queues. On the other hand, since two modes can possibly serve the downlink queues, the downlink service rates ϕ_i^{d} in (3.12) are only approximates of those in (3.21b) and (3.21c). In fact, it appears that ϕ_i^{d} given by (3.12) underestimate those calculated from (3.21b) and (3.21c). Note that from these definitions, a capacity region can be defined by choosing the components of $\boldsymbol{p}$ such that the inequalities (3.10) and (3.11) are satisfied. This capacity region appears to be a superset of the one described by (3.14) as can be observed in Figure 3.6, which has been obtained for Scenario 2 using the parameters in Table 3.3.

3.4.8 Numerical results

We study the performance of different policies discussed in Section 3.4.5 against each other through simulation. The performance of a policy is measured as the ratio of the average flow-level delay using the dynamic policy to the same when the optimal static policy is used. We define various scenarios by fixing the service rates of the different modes.

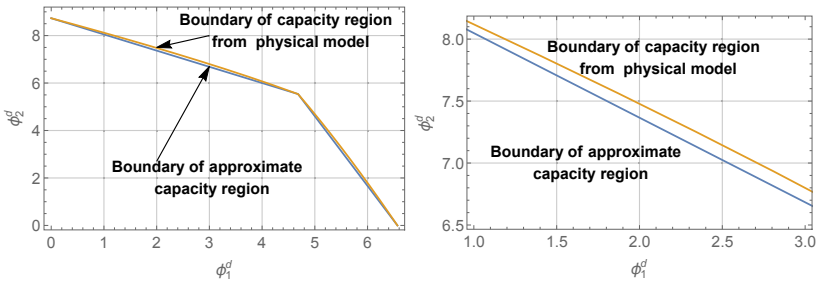


Figure 3.6. The capacity region from the physical model and the same obtained from (3.14) when $p^{\mathrm{uu}} = 0.5$ for Scenario 2 described in Section 3.4.8. Note that the difference is discernible only in the magnified picture (seen in the right panel). We can also see that the capacity region obtained from the physical model is clearly a superset of the one approximated by (3.14).

Table 3.2. Summary of Scenario 1

Scenario	Service rates $\mu_1^{\mathrm{dd}} = \mu_2^{\mathrm{dd}} = 3,$ $\mu_1^{\mathrm{du}} = \mu_2^{\mathrm{ud}} = 5$	Arrival rates
1a	$\mu_1^{\mathrm{uu}} = \mu_2^{\mathrm{uu}} = 7$	$\lambda_1^{\mathrm{d}} = \lambda_2^{\mathrm{d}}, \lambda_2^{\mathrm{u}} = 0$
1b	$\mu_1^{\mathrm{uu}} = \mu_2^{\mathrm{uu}} = 1$	$\lambda_1^{\mathrm{u}} = \lambda_2^{\mathrm{u}}, \lambda_2^{\mathrm{d}} = 0$

In Scenario 1, we study the performance of the different priority based policies by explicitly fixing the values of service rates μ_i^{δ} and assuming that the service times are symmetric (3.9) and exponentially distributed. These results are reported in Publication IV and we present here the cases that demonstrate performance of the stochastic optimality results. For this, we choose two Subscenarios 1a and 1b, which have their service rates fixed in such a way that HU and HD are stochastically optimal under the appropriate arrival rates at the uplink and the downlink, respectively, as seen from the results in Section 3.4.6.

To vary the traffic, we change the arrival rates so that the uplink load (ρ^{u}) and the downlink load (ρ^{d}) are the same and the total load, $\rho = \rho^{\mathrm{u}} + \rho^{\mathrm{d}}$, is varied from 0 to 1, where ρ^{u} and ρ^{d} are defined as

$$\rho^{\mathrm{u}} = \frac{\max\{\lambda_1^{\mathrm{u}}, \lambda_2^{\mathrm{u}}\}}{\mu_1^{\mathrm{uu}}}, \quad \rho^{\mathrm{d}} = \frac{\min\{\lambda_1^{\mathrm{d}}, \lambda_2^{\mathrm{d}}\}}{\mu_1^{\mathrm{dd}}} + \frac{|\lambda_1^{\mathrm{d}} - \lambda_2^{\mathrm{d}}|}{\mu_1^{\mathrm{du}}},$$

which naturally follow from (3.16) and (3.17) for these scenarios with symmetric base stations.

The performance of Scenarios 1a and 1b is shown in Figure 3.7. In Scenario 1a (Figure 3.7, left panel), RP is consistently providing 60% or more gain compared to the optimal static policy, and both RP and BF are better

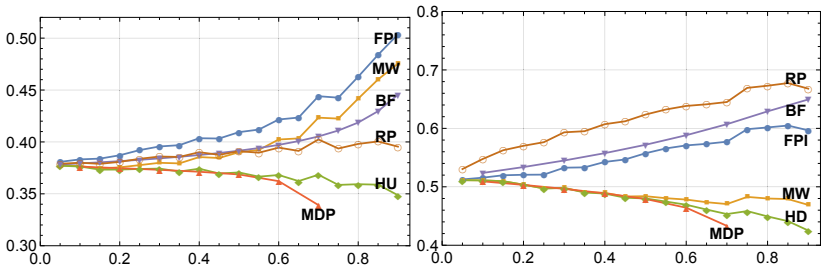


Figure 3.7. Performance of Scenario 1a (left) and Scenario 1b (right). The vertical axes represent the ratio of the average delay using the dynamic policy to the same using the optimal static policy while the horizontal axes denote the system load.

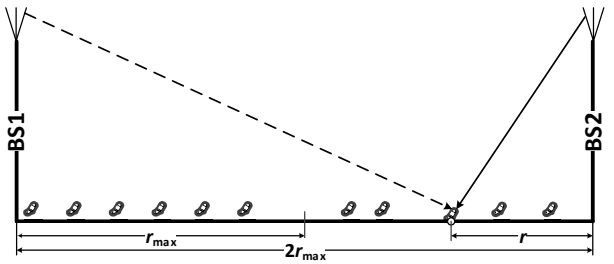


Figure 3.8. Linear sectorized network with two base stations.

than either of MW or FPI. However, the performance of all these policies is not anywhere close to that of the optimal policy, HU, especially at higher loads. In Scenario 1b (Figure 3.7, right panel), RP and BF are the worst performing dynamic policies while MW performs almost as good as the optimal dynamic policy for $\rho \leq 0.5$. In both these scenarios, the stochastically optimal dynamic policy is known, which is verified by the policy iteration algorithm (the MDP policy in the figure) when the values of load are small. At higher values of the load, the MDP policy suffers from truncation errors and underestimates the average delay as explained earlier.

In Scenario 2, we determine the service rates using a physical channel model, discussed in Section 3.4.7, in a linear network with two base stations which are $2r_{\max}$ distance apart as shown in Figure 3.8. By using (3.20) and the Shannon capacity theorem, the mean service times are calculated as

Table 3.3. Summary of Scenario 2

Physical parameters	Service completion rates
$\alpha = 3$, $r_{\max} = 400$ [m],	$\mu_1^{\mathrm{uu}} = 1.07$ [1/s], $\mu_2^{\mathrm{uu}} = 2.09$ [1/s],
$C_{\max}^{\mathrm{u}} = 50$ [Mbps],	$\mu_1^{\mathrm{dd}} = 9.36$ [1/s], $\mu_2^{\mathrm{dd}} = 11.10$ [1/s],
$C_{\max}^{\mathrm{d}} = 150$ [Mbps], $P^{\mathrm{ue}} = 0.2$ [W],	$\mu_1^{\mathrm{du}} = 13.10$ [1/s], $\mu_2^{\mathrm{ud}} = 17.50$ [1/s]
$P^{\mathrm{bs}} = 20$ [W],	
$I_1^{\mathrm{uu}}/2 = I_2^{\mathrm{uu}} = I_1^{\mathrm{dd}}/2 = I_1^{\mathrm{dd}} =$	
$I_1^{\mathrm{du}}/2 = I_2^{\mathrm{ud}} = 1.0 \times 10^{-7}$ [W],	
$B_1^{\mathrm{u}} = B_2^{\mathrm{u}} = B_1^{\mathrm{d}}/3 =$	
$B_2^{\mathrm{d}}/3 = 100$ [KiB]	

$$1/\mu_i^{\mathrm{uu}} = \mathbf{E}[B_i^{\mathrm{u}}]\,\mathbf{E}\left[\frac{1}{c_i^{\mathrm{uu}}(R)}\right] = \int_0^{r_{\max}} \frac{r_{\max}^{-1}\,\mathbf{E}[B_i^{\mathrm{u}}]\;\mathrm{d}r}{\min\left\{W\log_2\left(1+\frac{P^{\mathrm{ue}}/r^{\alpha}}{I_i^{\mathrm{uu}}}\right), C_{\max}^{\mathrm{u}}\right\}},$$

$$1/\mu_1^{\mathrm{du}} = \mathbf{E}[B_1^{\mathrm{d}}]\,\mathbf{E}\left[\frac{1}{c_1^{\mathrm{du}}(R)}\right] = \int_0^{r_{\max}} \frac{r_{\max}^{-1}\,\mathbf{E}[B_1^{\mathrm{d}}]\;\mathrm{d}r}{\min\left\{W\log_2\left(1+\frac{P^{\mathrm{bs}}/r^{\alpha}}{I_1^{\mathrm{du}}}\right), C_{\max}^{\mathrm{d}}\right\}},$$

$$1/\mu_2^{\mathrm{ud}} = \mathbf{E}[B_2^{\mathrm{d}}]\,\mathbf{E}\left[\frac{1}{c_2^{\mathrm{ud}}(R)}\right] = \int_0^{r_{\max}} \frac{r_{\max}^{-1}\mathbf{E}[B_2^{\mathrm{d}}]\;\mathrm{d}r}{\min\left\{W\log_2\left(1+\frac{P^{\mathrm{bs}}/r^{\alpha}}{I_2^{\mathrm{ud}}}\right), C_{\max}^{\mathrm{d}}\right\}},$$

$$1/\mu_i^{\mathrm{dd}} = \mathbf{E}[B_i^{\mathrm{d}}]\,\mathbf{E}\left[\frac{1}{c_i^{\mathrm{dd}}(R)}\right] = \int_0^{r_{\max}} \frac{r_{\max}^{-1}\,\mathbf{E}[B_i^{\mathrm{d}}]\;\mathrm{d}r}{\min\left\{W\log_2\left(1+\frac{P^{\mathrm{bs}}/r^{\alpha}}{\frac{P^{\mathrm{bs}}}{(2r_{\max}-r)^{\alpha}}+I_i^{\mathrm{dd}}}\right), C_{\max}^{\mathrm{d}}\right\}}$$

where W denotes the bandwidth of the system, α the path loss exponent, $C_{\max}^{\mathrm{u}}$ and $C_{\max}^{\mathrm{d}}$ the up/downlink maximum rates, P^{ue} and P^{bs} the transmission powers of the user equipment and the base station, and finally I_i^m the external interference. The capacity functions reflect our interference modeling assumptions: in the 'uu' mode and 'ud'/'du'-modes the other base station is not causing additional interference and only external interference I limits the rates, but in the 'dd' mode, the other base station is causing additional interference (dashed arrow) from distance $2r_{\max} - r$. Note that in this scenario, each job is served at a rate that is based on its location in the cell and the service times are non exponential, unlike in Scenario 1, where the service times were exponentially distributed.

We set the values as shown in Table 3.3 and obtain the values of the corresponding service rates.

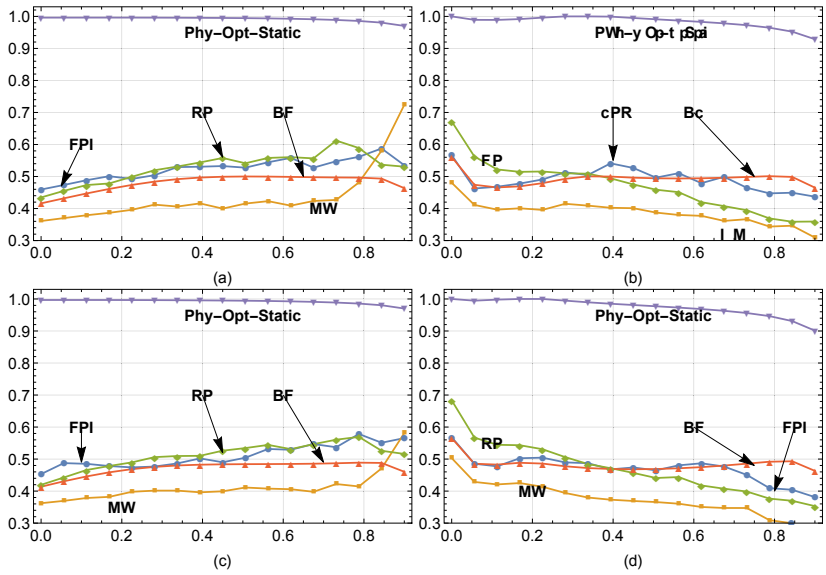


Figure 3.9. Performance of Scenario 2 where the panels (a), (b), (c) and (d) represent the Scenarios 2a, 2b, 2c and 2d, respectively. The vertical axes represent the ratio of the average delay using the dynamic policy to the same using the optimal static policy while the horizontal axes denotes the fraction of maximum possible λ_i^δ that can be stabilized where the pair (i, δ) is $(1, \mathrm{u}), (1, \mathrm{d}), (2, \mathrm{u})$ and $(2, \mathrm{d})$ for panels (a), (b), (c) and (d), respectively.

For this scenario, we choose the nominal values of the arrival rates as $\lambda_1^{\mathrm{u}} = 0.2, \lambda_1^{\mathrm{d}} = 2, \lambda_2^{\mathrm{u}} = 0.5, \lambda_2^{\mathrm{d}} = 3.2\,[1/\mathrm{s}]$. In Scenario 2a, we increase λ_1^{u} from 0 until the stability conditions (3.16) and (3.17) are satisfied while the rest of the arrival rates are held to their nominal values. In Scenarios 2b, 2c and 2d, only λ_1^{d}, λ_2^{u} and λ_2^{d}, respectively are changed similarly while the rest of the arrival rates are set to the nominal values.

In Scenario 2a, MW is the best performing policy at low loads. At high load value, the uplink queue length becomes very large and the performance of MW suffers. Throughout the load region BF consistently provides a gain of 50 % or more over the optimal static policy while RP and FPI perform slightly worse than BF.

In Scenario 2b, as shown by Figure 3.9 (b), we again observe that MW is again the best performing policy throughout the load region while FPI and BF provide about 50% gain over the optimal static policy throughout the load region. The performance of RP becomes better with increasing load and at higher values of load, it provides as much as 65% gain over the optimal static policy.

In Scenario 2c (Figure 3.9 (c)), the performance of the policies is almost identical to Scenario 2a (Figure 3.9 (a)), except for MW at the highest load

values as μ_2^{uu} is relatively higher than μ_1^{uu}. In Scenario 2d (Figure 3.9 (d)), we observe the policies perform almost identically as in Scenario 2b (Figure 3.9 (b)).

Note that we have used the approximate service rates as defined in (3.12) instead of the rates, (3.21), obtained from the physical model to calculate the optimal static delays. We also plot the ratio of the delay from physical model rates to the ones obtained from the approximated values for the optimal static policy which are identified by the label 'Phy-Opt-Static' in Figure 3.9. The physical model values are obtained by conducting a local search around the approximated values. The approximated values tend to underestimate the service rates of the queues, and therefore the average queue lengths are smaller when the physical model service rates are used. The approximation error tends to grow with increasing load values and appears to be very small when the fraction of the arrival rate that can be stabilized is less than 0.8 in the scenarios considered.

3.5 Summary

In this chapter, we considered the intercell coordination schemes for two neighboring base stations at the flow level. We have considered two categories of users which need to be scheduled by the base station. The objective in all cases is to minimize the flow-level delay so that the average throughput of the system can be maximized.

The first category of users is identified by the location from the base station, which affects their service rate. Based on the information about the traffic parameters, the service rate and the queue length, we have devised three policies to run the stations. Under the low interference conditions, we have observed that, a simple dynamic policy that uses both the stations as much as possible, provides good performance. In most of the cases, this policy is better than the one obtained by applying the systematic improvement algorithm, policy iteration, from MDP theory.

In the second case, the users are classified by their transmission direction in a dynamic TDD setting, and the service rates of the different queues are coupled with each other. We have identified the maximal stability region of such system. Even in such complex systems, we discovered that simple priority policies can be stochastically optimal, albeit only by imposing severe restrictions on system parameters. However, if the conditions for the stochastic optimality are not met, then the priority policies

become bad very quickly and may not even be maximally stable. The policies that are not based on priority, like MW and RP, show very good performance relative to the optimal static policy, even when more realistic channel models are used. Thus, while designing future wireless systems, these simple state-dependent scheduling policies may be used to provide good system throughput.

4. Opportunistic scheduling in wireless systems

4.1 Introduction

In a cellular wireless system, due to different path-loss and fading phenomena, the downlink transmission power at the user rapidly changes at different locations at various times. This leads to large variation in the rate of downlink data transmission by the base station, which typically operates in very short timeslots. By harnessing the information received from the users in the form of Channel Quality Index (CQI), the base station can predict the data rate that a user can support and then serve it accordingly. Thus, if there are many users in the system, the base station can schedule the data transfer to the user with a relatively good channel at any time.

Opportunistic scheduling is the method by which the station can schedule data transmission to users by taking their channel conditions into account. The station can exploit the multiuser diversity by serving the users with the best channel conditions, and thus, achieve the opportunistic scheduling gain. In practice, fair opportunistic schedulers are frequently used, that do not starve the users with bad channel conditions. For example, the Proportionally Fair (PF) scheduler has nice fairness properties and is widely used.

The opportunistic schedulers should operate at the same time scale in which the channel variation occurs. However, when dealing with the elastic traffic, i.e, TCP-controlled file transfers, it is again more worthwhile to consider the problem at the flow level. The arrival and the departure of the service demands from the users, called the flow-level dynamics, is typically much slower than the flow scheduling rate.

Indeed, if the time scales between the scheduling rate and the flow dy-

namics are separated by some orders of magnitude (time-scale separation), we can abstract the average long-term service received by the users from an opportunistic scheduler as the points in the so-called capacity region. Recall from Chapter 3, capacity region is the set of all possible long-term service rates with which the users can be served by the station. The capacity region depends on the number of users and the channel statistic of each user. We can define an opportunistic scheduler by properly selecting the operating point in the capacity region so that some system parameter is optimized. These rates can then be implemented using some weight based scheduler.

Besides the geometry of the capacity region, other information, such as the size of the flow, can also be exploited to find an optimal operating point in the capacity region. An example of utilization of the size information of the flows in a non-opportunistic system, such as the M/G/1 queue, is the so-called *shortest-remaining-processing-time* (SRPT) scheduling policy, which leads to the minimum flow-level delay [80] among all possible scheduling policies.

The opportunistic gain, however, is completely lost when a purely SRPT-like policy is used to schedule users in time-varying channels. On the other hand, we observe that the rate of increase of opportunistic gain typically slows down with the increasing number of users. Thus, when enough users are simultaneously served, it could be worthwhile to harness the gain from an SRPT-like policy, even if it means giving up the additional opportunistic gain. This can be done, e.g., by serving only a subset of users in the system that have the smallest remaining processing times. The scheduler could also provide more service timeslots to the smaller flows compared to the larger flows, even if the former have relatively worse channel conditions. This could lead to an SRPT-FM-like policy that is optimal in a transient system (i.e., a system where there are no new arrivals) with many heterogeneous servers [69, 79]. These issues are primarily studied in Publication VII. Then in Publication VIII, we discuss the way such flow level-policies can be implemented at the timeslot level, at which the system actually works. The optimality results derived for the transient cases are then applied as heuristics in a system with dynamic flow arrivals in both the publications.

At the timeslot level, for a transient case, the opportunistic scheduling problem can also be formulated as a *restless-multiarmed-bandit* (RMAB) problem. Whereas the earlier flow-level approach was applicable only to

the homogeneous channels, where all users have identical channel processes, the RMAB approach can also tackle the heterogeneity of the users' channels. Although the original RMAB cannot be solved exactly, by relaxing the constraints of the problem, and following the so-called Whittle index approach [98], the problem can potentially be solved. This solution leads to the heuristic policies that are applicable even to the original scheduling problem in the asymmetric case with heterogeneous channels. This study is performed as a part of Publication IX.

4.2 Related work

The idea of scheduling users based on their channel conditions is as old as wireless system. For modern wireless systems such as Evolution-Data Optimized (EVDO)/High Data Rate (HDR), it is shown [50, 14] that a certain channel-aware opportunistic scheduling algorithm leads to Proportionally Fair (PF) allocation as defined in [51]. PF is an α-fair [63] scheduler, which exploits the channel information. Similar rate-based schedulers have been discussed in [15, 18, 67], which consider only the channel state information while making scheduling decisions. Other opportunistic schedulers, such as the Max-Weight scheduler [92], utilize service rate and the queue length. The Max-Weight scheduler has been shown to be throughput optimal (maximally stable) in [86].

In [28], we see that with the time-scale separation assumption, processor-sharing abstraction can be used to model PF scheduling in opportunistic systems. Moreover, capacity region is also introduced, which is the collection of the long-term throughputs achieved by all possible opportunistic policies. In fact, the time-scale separation and flow level models have been used in a number of works to study the performance of various scheduling policies [19, 28, 79]. The policies based on the utility functions of the throughput have been shown to be maximally stable in [30], while rate-based schedulers could get unstable quite quickly [6]. However, giving priority to the user which has its best possible channel has been observed to be maximally stable [6]. This has been proved for i.i.d. arrivals and geometric flow sizes for i.i.d. channels in [13], and for Markovian channels in [52].

The SRPT queueing discipline is shown to be optimal in [80]. This classic result is applicable only for a single server system working at a constant rate. In a transient system with heterogeneous servers, SRPT-FM

(shortest-remaining-processing-time-to-fastest-machine) policy is known to be optimal [69]. In [79], by assuming the capacity region to be a nested polymatroid, the authors show that the optimal scheduling policy allocates the service rates to the flows in the capacity region that follow the SRPT-FM principle.

In a transient system, the timeslot level optimal scheduling problem can be formulated as a dynamic program that can be numerically solved [93]. Moreover, [12] formulates the problem of opportunistically scheduling a user in a time varying channel as the restless multi-armed bandit (RMAB) allocation problem. RMAB is a generalization of the classic multi-armed bandit problem and has been introduced in [98]. It has been shown [12] that if the scheduling constraint, which requires only one user to be served in one timeslot, is replaced by a looser constraint that only one user on *average* needs to be served, then the Lagrangian version of the problem becomes separable. This may lead to the solution of the relaxed problem, if it is indexable. This, in turn, can provide a good heuristic to solve the original problem.

In [12], the relaxed opportunistic scheduling problem has been shown to be indexable for geometric job sizes and i.i.d. channels, and in [13] the corresponding Whittle-index based policy has been proven to be maximally stable and fluid optimal. In [49], the problem is generalized to a Markovian channel with two states.

In [31], the authors provide good index-based policies for a Markovian channel with multiple states and geometric job sizes. In [89], similar policies are derived for non-geometric job sizes in i.i.d. channels with two states, and in [90], the results are extended to i.i.d. channels with multiple states. In all three works, however, the authors do not prove that the problem is indexable and only consider the non-anticipating policies, which are not aware of the exact size of the remaining job when the decisions are made. The non-anticipating policies, instead, use the information about the attained service.

4.3 Size-aware opportunistic scheduling problem

We first define the opportunistic scheduling problem in a transient system, and then continue to discuss the strategies to solve the problem by taking into account the sizes of the jobs to be scheduled. In our numerical studies, these strategies are used as heuristics applied to a dynamic

system, where they are tested against other known scheduling policies.

We consider a timeslotted system where each timeslot is τ units long and they are indexed by $t = 0, 1, \ldots$. In the beginning (at $t = 0$), there are K flows[1] in the system. The system is said to be *transient* or *static* if new flows do not arrive after $t > 0$. Otherwise, the system is said to be *dynamic*. In the sequel, we focus exclusively on the transient systems starting with K users. When we study a dynamic system, it is explicitly stated.

The channel state of a user i at the beginning of timeslot time t is denoted by $R_i(t)$, where $R_i(t)$ is a stationary stochastic process associated with user i and indexed by t, which takes values in $\mathcal{R}_i \subset \mathbb{R}_+$. We assume that user i, if scheduled, can be served at rate $R_i(t)$ during time interval $[t, t+\tau)$. Furthermore, $R_i(t)$ are assumed to be independent across users i. The channels are *homogeneous* (*symmetric* setting) if $R_i(t)$ have identical distributions for all users i. We collect the channels of all the users as the components of $\boldsymbol{R}(t) = (R_1(t), \ldots, R_n(t))$ when there are n users in the system.

In timeslot t, an *opportunistic scheduler* schedules user i, to be served at rate $R_i(t)$, taking into account the channel states of all n users at that time. Such schedulers exploit the multiuser diversity, and therefore can potentially provide good scheduling gain. We define the *opportunistic gain* γ_n as

$$\gamma_n = \mathbf{E}[\max\{R_1(t), \ldots, R_n(t)\}] \tag{4.1}$$

when there are n users in the system. An *opportunistic scheduling policy* (or *scheduling policy*) π is a well defined rule, using which the scheduler selects only one user to schedule in timeslot t. Let Π_n be the set of all such policies when there are n users. If a user i is scheduled under a policy $\pi \in \Pi$ in timeslot t, we define $A_i^\pi(t) = 1$. Otherwise $A_i^\pi(t) = 0$. Clearly, since only one user can be served in one timeslot

$$\sum_{i=1}^{K} A_i^\pi(t) \leq 1, \tag{4.2}$$

where the equality is achieved in a non-idling system. Moreover, we assume that the traffic is elastic and the data buffer is always non-empty so that the station can schedule any user it likes.

We further assume that at $t = 0$, user i has a job of size x_i (in bits). As the users are served according some policy π, the job sizes decrease (or

[1] In this chapter, 'jobs', 'users' and 'flows' are used interchangeably.

remain the same) as t increases. We denote the remaining size of jobs at time t by $Y_i^\pi(t)$. Thus, if a user j is scheduled in timeslot t, its job size decreases by $R_j(t)\tau$ bits, while the remaining sizes of all other jobs stay unchanged, i.e.,

$$A_j^\pi(t) = 1 \Rightarrow Y_i^\pi(t+1) = \begin{cases} \max(Y_i^\pi(t) - R_i(t)\tau, 0), & \text{if} \quad i = j, \\ Y_i^\pi(t), & \text{if} \quad i \neq j. \end{cases}$$

A user i exits the system at time t if and only if $A_i^\pi(t) = 1$ and $R_i^\pi(t)\tau \geq Y_i^\pi(t)$, and therefore is no longer scheduled after that. Before it exits, user i accrues a holding costs at rate c_i. The average of the total holding cost of all users is then given by

$$\mathbf{E}\left[\sum_{t=0}^{\infty}\sum_{i=1}^{K} c_i \mathbf{1}_{\{Y_i^\pi(t)>0\}}\right]. \tag{4.3}$$

Our *original problem* (OP) is to find an opportunistic scheduling policy π^* which minimizes (4.3), that also utilizes the size information $Y_i^\pi(t)$, besides the channel information $R_i(t)$, so that the scheduling constraint (4.2) is satisfied.

4.4 Homogeneous user population

For a the symmetric setting with homogeneous channels and unity holding cost for all users ($c_i = 1$ for all i), OP becomes considerably simpler. We can also use the flow-level models to arrive at optimal operating points that can then be mapped to the optimal timeslot-level scheduling policy.

4.4.1 Flow-level problem description

We begin by presenting the flow-level version (FP) of OP described in Section 4.3. We assume that the traffic is elastic and the decisions about scheduling the flows are made much more frequently than the traffic arrives to and departs from the system (*time-scale separation*) as in Chapter 3. This allows to use the long-term throughput of a user, instead of instantaneous service rates, for analyzing the system. The problem of determining a good scheduling policy that minimizes the holding costs at the timeslot level is now manifested as the one of finding the operating points in the *capacity region*, which is the set of all the possible long-term service rates at which the different flows are served using an opportunistic scheduler.

When there are n user in the system, in each timeslot the channel state of user i takes some value in finite set $\mathcal{R}_i$, i.e., $\boldsymbol{R}(t)$ takes values in $\mathcal{R} = \mathcal{R}_1 \times \cdots \times \mathcal{R}_n$, and $p(\boldsymbol{r}) = \mathbf{P}\{\boldsymbol{R}(t) = \boldsymbol{r}\}$, where $\boldsymbol{r} = (r_1, \ldots, r_n) \in \mathcal{R}$. The long-term average service rate of user i is represented by $c_{n,i}$ when there are n users in the system.[2] The set $\mathcal{C}_n \in \mathbb{R}_+^n$, which is the set of all achievable long-term data rate vectors, is given by [28]

$$\mathcal{C}_n = \{\theta = (\theta_1, \ldots, \theta_n) \in \mathbb{R}_+^n : z(\theta) \geq 1\}, \tag{4.4}$$

where $z(\theta)$ is the optimal value of the following linear program

$$\begin{aligned} &\max\ z \\ &\text{s.t.}\quad z \leq z_i = \sum_{\boldsymbol{r} \in \mathcal{R}_n} \frac{p(\boldsymbol{r}) x_i(\boldsymbol{r}) r_i}{\theta_i}, \quad i \in \{1, \ldots, n\} \\ &\qquad \sum_{i=1}^{n} x_i(\boldsymbol{r}) \leq 1, \quad \boldsymbol{r} \in \mathcal{R}_n, \\ &\qquad x_i(\boldsymbol{r}) \geq 0, \quad i \in \{1, \ldots, n\}. \end{aligned}$$

Here, $\mathcal{C}_n$ is called the *opportunistic capacity region* or the *capacity region*. The capacity region is *symmetric* if, for any vector $\boldsymbol{c} \in \mathcal{C}$, any other vector, whose components are a permutation of the components of $\boldsymbol{c}$, is also in $\mathcal{C}$. From the complementary slackness condition of the above linear program, we see that by properly choosing the weight vector $\boldsymbol{w} = (w_1, \ldots, w_n) \neq (0, \ldots, 0)$, where w_i is the weight for flow i, and allocating the timeslot t to user

$$i^* = \arg\max_i w_i R_i(t),$$

any non-dominated rate vector in the capacity region as the long-term service rate can be achieved.[3] Such schedulers are called *weight-based schedulers*.

When there are K jobs in the system, whose initial sizes are x_i for user $i \in \{1, \ldots, K\}$, the indexing is done in the descending order of their sizes. We choose a point $\boldsymbol{c}_K = (c_{K,1}, \ldots, c_{K,K}) \in \mathcal{C}_K$, whose component $c_{K,i}$ is the long-term rate user i is served by the system. Thus, the job i decreases at rate $c_{K,i}$, and when any user departs from the system, the remaining $K-1$ jobs are again reindexed based on their sizes, and a new operating

[2]Note that in this section, we assume the holding costs of the jobs are all unity, the symbol c is now used to denote the operating points in the capacity region. This is done so that the notation is in line with what we have used in Publications VII and VIII.

[3]If $\arg\max_i w_i R_i(t)$ is not unique, a well defined tie-breaking rule has to be used to select just one flow.

point $\boldsymbol{c}_{K-1}$ is chosen. The jobs are now denoted by $x_{K-1,1}, \ldots, x_{K-1,K-1}$ with $x_{K-1,1} \geq \cdots \geq x_{K-1,K-1}$. Thus, the vector of the original job sizes is $(x_1, \ldots, x_K) = (x_{K,1}, \ldots, x_{K,K})$. We define a vector of operating points $\phi = (\boldsymbol{c}_K, \ldots, \boldsymbol{c}_1) \in \mathcal{C}_K \times \cdots \times \mathcal{C}_1$ as an operating policy, and Φ as the set of all such policies. Since each job accrues a holding cost at rate 1 as long as it remains in the system, the total holding cost (total flow-level delay) of all jobs under an operating policy ϕ is

$$T^{\phi} = \sum_{i=1}^{K} T_i^{\phi}, \tag{4.5}$$

where T_i^{ϕ} is the time job i stays in the system under policy ϕ. In FP, we seek an operating policy $\phi^* \in \Phi$ that minimizes T^{ϕ} utilizing the size information $x_{i,j}$, where $i \in \{1, \ldots, K\}$ and $j \in \{1, \ldots, i\}$, i.e.,

$$\phi^* = \arg\min_{\phi \in \Phi} T^{\phi}. \tag{4.6}$$

This operating policy ϕ^* then has to be mapped to a scheduling policy π^*, which can then be actually implemented at the timeslot level to opportunistically schedule the users.

4.4.2 Optimization at the flow level

We begin by solving the FP in a symmetric setting for users with homogeneous channels, where $\mathcal{C}_n$ has the following properties:

- $\mathcal{C}_n$ is a compact subset of $\mathbb{R}^n$, i.e., $\mathcal{C}_n$ is closed and bounded.
- $\mathcal{C}_n$ is symmetric, i.e., if $\boldsymbol{c} \in \mathcal{C}_n$, then any permutation $\tilde{\boldsymbol{c}}$ of its components is also in $\mathcal{C}_n$.
- $\mathcal{C}_n$ is convex.

Note that the capacity region abstracts the opportunistic nature of the scheduling policy, and the best trade-off between size-based and the opportunistic scheduling can be achieved by the optimal operating policy which selects a sequence of operating points in $\mathcal{C}_n$.

Let $g_1, \ldots, g_K$ be a sequence of real valued functions with $g_k(\boldsymbol{c}_k)$ defined on $\mathcal{C}_k$, $G_1^*, \ldots, G_K^*$, be a sequence of scalars, and $\boldsymbol{c}_1^*, \ldots, \boldsymbol{c}_K^*$ be a sequence

of operating points with $\boldsymbol{c}_k^* \in \mathcal{C}_k$, which are defined recursively as follows:

$$g_1(\boldsymbol{c}_1) = \frac{1}{c_1}, \qquad G_1^* = \min_{\boldsymbol{c}_1 \in \mathcal{C}_1} g_1(\boldsymbol{c}_1), \quad g_1(c_1^*) = G_1^*, \tag{4.7}$$

$$g_k(\boldsymbol{c}_k) = \frac{1}{c_{k,k}} \left(k - \sum_{j=1}^{k-1} c_{k,j} G_j^* \right), \quad G_k^* = \min_{\boldsymbol{c}_k \in \mathcal{C}_k} g_k(\boldsymbol{c}_k), \quad g_k(\boldsymbol{c}_k^*) = G_k^*, \tag{4.8}$$

for all $k \in \{1, \ldots, K\}$. The existence of the minimum values of $g_k(\cdot)$ is guaranteed by the compactness property. Moreover, for $g_k(\cdot)$ and G_k^* to be well defined, it is sufficient that the capacity regions $\mathcal{C}_k$ be nested. Then in Theorem 2 of Publication VII, we solve FP as follows:

If the capacity regions $\mathcal{C}_1, \ldots, \mathcal{C}_K$ are such that $G_1^* < \cdots < G_K^*$, then the optimal operating policy is $\pi^* = (\boldsymbol{c}_1^*, \ldots, \boldsymbol{c}_K^*)$ for all sizes $x_{K,1} \geq \cdots \geq x_{K,K}$. The cumulative delay T^{π^*} satisfies

$$T^{\pi^*} = \sum_{k=1}^{K} x_{K,k} G_k^*. \tag{4.9}$$

In addition, $c_{k,j+1}^* \geq c_{k,j}^*$ for all $k = 2, \ldots, K$ and $j = 1, \ldots, k-1$ so that the optimal policy applies the SRPT-FM principle.

From this result we can see that to calculate the optimal operating points, the exact job sizes are not necessary; the share of the service rates for different jobs are determined based solely on the sizes of the jobs relative to each other. Also, the optimization is carried out over a single variable in each step, which makes the process relatively simple, computationally. Finally, as seen in (4.9) G_k^* represents delay cost per unit size for flow k under the optimal operating policy.

Moreover, a sufficient condition for the existence of such an optimal policy is satisfied if we define a capacity region as follows. Let π be the policy in which the opportunistic scheduler selects flow i for transmission with probability $p_i^\pi(\boldsymbol{r})$ when the channel state $\boldsymbol{R}(t) = \boldsymbol{r} \in \mathcal{R}$. Let Π_n be the set of all such scheduling policies when there are n flows. The long-term throughput for user i under scheduler π is

$$\theta_i^\pi = \sum_{\boldsymbol{r} \in \mathcal{R}^n} r_i p_i^\pi(\boldsymbol{r}) \, \mathbf{P}\{(R_1(t), \ldots, R_n(t)) = \boldsymbol{r}\}. \tag{4.10}$$

Then, the opportunistic capacity region, defined as the set of all feasible throughput vectors, is given by

$$\mathcal{C}_n = \{(\theta_1^\pi, \ldots, \theta_n^\pi) \in \mathbb{R}_+^n : \pi \in \Pi_n\} \tag{4.11}$$

This definition of the capacity region is equivalent to (4.4). As proved in Proposition 3 of Publication VIII, with this capacity region, we have, $G_1^* <$

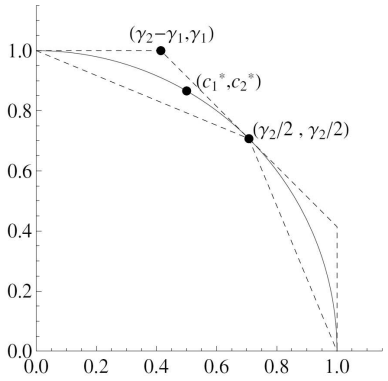


Figure 4.1. The capacity region for $n = 2$ flows. The solid line represents the α-ball region with $\alpha = 2$, while the inner and the outer dashed lines represent the OPS-limited polytope and polymatroid capacity regions.

$\cdots < G_K^*$, implying that an optimal operating policy can be determined by the preceding result.

When the capacity regions have explicit expressions, these results are used to obtain the optimal operating policy ϕ^*. In Publication VII, we have defined three such capacity regions—symmetric and nested polymatroids, symmetric and nested OPS-limited polytopes and α-ball capacity regions—for which the optimal operating policy ϕ^* has been characterized.

For a simple case of just two users, these capacity regions are depicted in Figure 4.1. We see that the α-ball capacity region, $c_1^\alpha + c_2^\alpha \leq 1$, whose border is shown by the solid line, is underestimated by the OPS-limited polytope capacity region, while the symmetric polymatroid capacity region overestimates it. Moreover, from the results in Publication VII, we obtain the rate vectors $\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right), \left(\sqrt{2}-1, 1\right)$ and $\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right)$, respectively, associated with the optimal operating policy for the α-ball, the symmetric polymatroid and the OPS-limited polytope capacity regions, respectively.

On the other hand, if the explicit expression of the capacity region is not available, a more direct approach to determine the optimal operating points by calculating G_k^* from the channel model of the users is shown in Corollary 1 of Publication VIII:

$$\begin{aligned} G_1^* &= \frac{1}{\gamma_1}, \\ G_k^* &= f_k^{-1}(k), \quad k = 2, \ldots, n, \quad \text{where} \\ f_k(a) &= \int_0^\infty \left(1 - \prod_{i=1}^{k-1} \mathbf{P}\{G_i^* R_i \leq r\}\, \mathbf{P}\{aR_k \leq r\}\ \mathrm{d}r\right). \end{aligned} \tag{4.12}$$

Here $f_k(a)$ is strictly increasing with $f_k(0) = k-1$ and $f_k(a) \to \infty$ as $a \to \infty$. Thus, the inverse function $f_k^{-1}(a)$ is well defined on $[k-1, \infty)$. This approach is used to calculate the optimal operating policy for the exponential and two-state channels as follows:

Exponential channel model

When the channel model is exponential, i.e., $R_i(t) \sim \text{Exp}(1)$, we do not have the explicit expression of the capacity region. The opportunistic gain γ_k when there are k users is given by

$$\gamma_k = \sum_{i=1}^{k} \frac{1}{i},$$

which is unbounded as $k \to \infty$. Thus, $f_k(a)$ is given by,

$$f_k(a) = \int_0^\infty \left(1 - (1 - e^{-\frac{r}{a}}) \prod_{i=1}^{k-1} \left(1 - e^{-\frac{r}{G_i^*}}\right)\right) \, \mathrm{d}r, \tag{4.13}$$

which is then inverted to calculate G_i^*. The corresponding optimal flow-level operating point $\boldsymbol{c}_k^*$ is then calculated as

$$c_{ki}^* = \int_0^\infty r e^{-r} \prod_{j \neq i} \left(1 - e^{-\frac{G_i^*}{G_j^*} r}\right) \, \mathrm{d}r. \tag{4.14}$$

Two-state channel model

In two-state channel models, we assume that the probability the channel takes one of two discrete values, r_1 and r_2 ($r_2 > r_1$), respectively, is

$$\mathbf{P}\{R_i(t) = r_1\} = 1 - p, \qquad \mathbf{P}\{R_i(t) = r_2\} = p. \tag{4.15}$$

The opportunistic gain γ_k is then given by

$$\gamma_k = r_1(1-p)^k + r_2(1 - (1-p)^k), \tag{4.16}$$

which converges to r_2 as $k \to \infty$. To compute G_k^*, we define $m(a), a > 0$, as the maximum $m \in \{0, \ldots, k-1\}$ for which $G_m^* r_2 < a r_1$, with the convention $G_0^* = 0$. Recursion (4.12) is now

$$f_k(a) = a(r_1(1-p)^{k-m(a)} + r_2 p) + r_2 \sum_{i=m(a)+1}^{k-1} G_i^* p (1-p)^{k-i}, \tag{4.17}$$

which can be used to find G_k^*. Let $m_k = m(G_k^*)$, using which the optimal operating point $\boldsymbol{c}_k^* = (c_{k,1}, \ldots, c_{k,k})$ is obtained as:

$$c_{k,i}^* = \begin{cases} 0, & i \leq m_k, \\ r_2 p (1-p)^{k-i}, & m_k < i < k, \\ r_1(1-p)^{k-m_k} + r_2 p, & i = k. \end{cases} \tag{4.18}$$

4.4.3 Solving OP at the timeslot level

The optimal operating points at the flow level can be easily mapped to a timeslot-level opportunistic scheduler π^*, which follows from Theorem 2 of Publication VIII stated below:

The optimal operating policy $\phi^* = (\boldsymbol{c}_1^*, \ldots, \boldsymbol{c}_K^*)$ can be implemented by a sequence of weight-based schedulers that, when there are k users in the system use a weight vector $\boldsymbol{w}_k = (G_1^*, \ldots, G_k^*)$ and break the ties by giving the timeslot t to the flow with the highest index i^* such that

$$G_{i^*}^* R_{i^*}(t) = \max\{G_1^* R_1(t), \ldots, G_k^* R_k(t)\}.$$

4.4.4 Numerical results

We now study the performance of the optimal policy in a transient system, TR-OPT, using simulations in a dynamic system. We assume that the new flows arrive according to a Poisson process of rate λ. We utilize a flow-level simulator as well as a packet level simulator for our studies, which serve the flows at the long-term service rate and the instantaneous rates, respectively. Naturally, if the time-scale separation assumption holds, then the two should provide identical performance.

We define three scheduling rules—PF, SRPT-P, OPS*—when there are n users as follows:

The *Proportionally Fair* (PF) operating point is defined as

$$\boldsymbol{c}^{\mathrm{PF}} = (\gamma_n/n, \ldots, \gamma_n/n)$$

which can be implemented at the timeslot level by the weight based scheduler

$$\boldsymbol{w}^{\mathrm{PF}} = (1, \ldots, 1).$$

The *SRPT-OPS(k)* operating point [79] is defined as

$$\boldsymbol{c}^{\text{SRPT-OPS}(k)} = \begin{cases} (0, \ldots, 0, \gamma_k/k, \ldots, \gamma_k/k), & k < n, \\ (\gamma_n/n, \ldots, \gamma_n/n), & k \geq n. \end{cases}$$

which is achieved by a weight based scheduler defined by

$$\boldsymbol{w}^{\text{SRPT-OPS}(k)} = \begin{cases} (0, \ldots, 0, 1, \ldots, 1), & k < n, \\ (1, \ldots, 1), & k \geq n. \end{cases}$$

It serves the $\min\{k, n\}$ smallest flows according to PF, and thus, both the vectors defined above have $\min\{k, n\}$ non-zero elements. It is possible to vary the performance of this rule by varying the value of k. When the value of k is chosen in such a way that the average delay is minimized using this policy, then this policy is referred to as OPS*.

The *SRPT-P* operating point [55] is defined as

$$\boldsymbol{c}^{\text{SRPT-P}} = (\theta_1^{\text{SRPT-P}}, \ldots, \theta_1^{\text{SRPT-P}}),$$

where

$$\theta_i^{\text{SRPT-P}} = \sum_{r \in \mathcal{R}} r \, \mathbf{P}\{R_1 \leq r\}^{i-1} \, \mathbf{P}\{R_1 = r\} \, \mathbf{P}\{R_1 < r\}^{n-i},$$

which prioritizes the flow with the highest instantaneous rate, with the ties are broken according to the SRPT principle.

The *TR-OPT* operating point, corresponding to the transient optimal policy, is defined as

$$\boldsymbol{c}^{\text{TR-OPT}} = (c_1^*, \ldots, c_n^*),$$

which is achieved by the weight based scheduler

$$\boldsymbol{w}^{\text{TR-OPT}} = (G_1^*, \ldots, G_n^*),$$

where c_i^* and G_i^* are defined in (4.7) and (4.8).

For the study, we consider the exponential channel model, the two-sate channel model, and the channel model taken from HDR specifications [12, Table 1] in Scenarios A, B and C, respectively. In the first two scenarios, the mean file sizes are exponentially distributed with mean 1 and timeslot duration of 0.001. In Scenario C, the flows are exponentially distributed with mean 50 KB, while the timeslot duration is 1.67 ms as specified in the specifications for HDR system. We define the performance of a policy as the mean number of flows under the policy, and the *performance ratio* of a policy as the ratio of the performance of a policy to the performance of PF under identical parameter setting.

Scenario A

For the scenario with the exponential channel model, the mean number of flows for different policies, and the performance ratio of TR-OPT are shown as a function of arrival rate λ in the left and the right panels of Figure 4.2, respectively. The continuous lines show the results from the flow-level simulator while the dots indicate the results from the timeslot-level simulator.

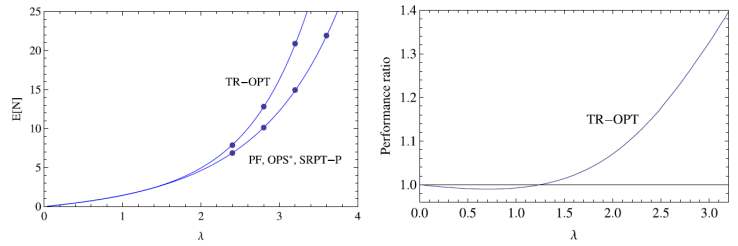


Figure 4.2. Mean number of flows with different policies (left) and ratio of the mean number of flows relative to PF (right) as a function of arrival rate for Scenario A.

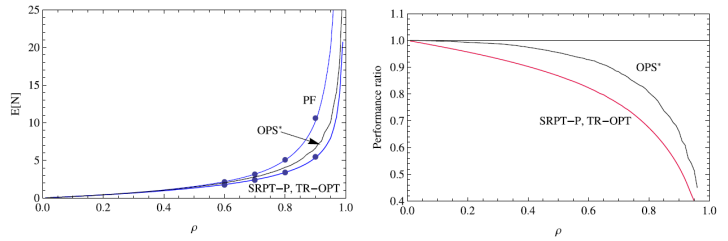


Figure 4.3. Mean number of flows with different policies (left) and ratio of the mean number of flows relative to PF (right) as a function of load for Scenario B.

PF and SRPT-P are identical in this scenario. Even OPS* provides almost the same (though, not identical) performance. Since the channel model is exponential, the opportunistic gain grows without bounds for PF. Consequently, the TR-OPT policy is becoming increasing worse than the rest of the policies, as the former cannot achieve the total service rate as seen from Publication VIII, Figure 2 (right panel). However, as evident from Figure 4.2 (right panel) TR-OPT is able to achieve slight gain over other policies at light loads.

Scenario B

For Scenario B, the mean number of flows for different policies and the performance ratios of these policies are shown as a function of load ρ in the left and the right panels of Figure 4.3, respectively. The load is defined as $\rho = \lambda \bar{X}/r_2$, where $\bar{X}$ is the mean size of the flows. The maximum capacity of the system is r_2 and the system is stable as long as $\rho < 1$. We take $r_1/r_2 = 1/10$ and $p = 0.5$.

The continuous lines show the results from the flow-level simulator while the dots indicate the results from the timeslot-level simulator. Clearly, SRPT-P and TR-OPT have almost identical performances, and they perform better than PF and OPS* throughout the load region.

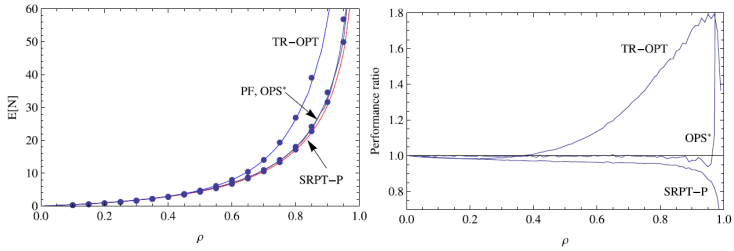


Figure 4.4. Mean number of flows with different policies (left) and ratio of the mean number of flows relative to PF (right) as a function of load for Scenario C.

Scenario C

For Scenario C, the mean number of flows for different policies, and the performance ratios of these policy are shown as a function of load ρ in the left and the right panels of Figure 4.4, respectively. The load ρ is defined in an identical way as in Scenario B, where r_2 is replaced by the maximum rate possible in the HDR system.

The continuous lines show the results from the flow-level simulator while the dots indicate the results from the timeslot-level simulator. Again we observe that SRPT-P performs better than all other policies, especially at high loads. The performance of TR-OPT degrades rapidly with increasing loads. The OPS* is almost the same as PF and at high loads, the performance degrades due to the local optimization of the k parameter.

4.5 Heterogeneous user population

A heterogeneous user population may have different channel models, holding costs or job size distributions. Even with the heterogeneity in the channel model alone, the capacity region becomes asymmetric. Thus, the problem is not solvable by the flow-level techniques of Section 4.4.2.

Therefore, we reformulate the problem, for a case of two-state channel model and job sizes approximated by the shifted Pascal distribution, as the restless-multiarmed-bandit (RMAB) problem in Section 4.5.1. We use the so-called Whittle index approach [98] to solve the problem. This relaxes the scheduling constraint making the problem separable for each user, and thus, tractable even to a heterogeneous user population. The solution obtained for the relaxed problem is applied as a heuristic in the original unrelaxed problem.

4.5.1 Relaxed opportunistic scheduling problem

We assume that the channels of the users are independent of each other, and take two discrete values with point probabilities

$$\mathbf{P}\{R_i(t) = r_{i,1}\} = 1 - q_i \qquad \text{and} \qquad \mathbf{P}\{R_i(t) = r_{i,2}\} = q_i, \tag{4.19}$$

with $r_{i,2} > r_{i,1}$ for user i. The mean value of channel state for user i is

$$\bar{r}_i = \mathbf{E}[R_i(t)] = (1 - q_i)r_{i,1} + q_i r_{i,2}. \tag{4.20}$$

We approximate each job size x_i by a random variable X_i that has J consecutive phases,

$$x_i \approx X_i = X_{i,1} + \cdots + X_{i,J},$$

and size $X_{i,j}$ of the jth phase of X_i is geometrically distributed,

$$\mathbf{P}\{X_{i,j} = n\} = (1 - p_i)^{n-1} p_i, \quad n \in \{1, 2, \ldots\},$$

$p_i = J/x_i$. Thus X_i has the so-called *shifted Pascal distribution* with

$$\mathbf{P}\{X_i = n\} = \frac{(n-1)!}{(n-J)!(J-1)!}(1 - p_i)^{n-J} p_i^J, \quad n \in \{J, J+1, \ldots\},$$

$$\mathbf{E}[X_i] = x_i \qquad \text{and} \qquad \frac{\mathbf{Var}[X_i]}{\mathbf{E}[X_i]^2} = \frac{1}{J} - \frac{1}{x_i}.$$

Note that the relative variance of random variable X_i is very small when x_i and J are large. Under these conditions, X_i, whose mean is x_i, provides a reasonably good approximation of the original job of size x_i. Moreover, if job i is scheduled at time t, then it is served at rate $R_i(t) = r$, and thus, has probability Jr/x_i of completing its current phase at the end of the timeslot.[4]

Let $Z_i^\pi(t)$ denote the remaining number of phases of a job i at time t under some policy π. Thus, the state of a job i under policy π at time t is described by the pair $(Z_i^\pi(t), R_i(t))$, which the scheduler takes into account while making the scheduling decision. The expected total cost under policy π is given by

$$f^\pi = \mathbf{E}\left[\sum_{t=0}^{\infty}\sum_{i=1}^{K} c_i \mathbf{1}_{\{Z_i^\pi(t)>0\}}\right], \tag{4.21}$$

which is the modified objective function to be minimized by selecting an appropriate policy π^*. This minimization problem is an RMAB problem if

[4] We assume $\tau = 1$ in the sequel.

the strict constraints (4.2) are used. However, as suggested in [98], when the same problem is considered under the relaxed scheduling constraints

$$g^{\pi} = \mathbf{E}\left[\sum_{i=1}^{K} A^{\pi}(t)\right] \leq 1, \tag{4.22}$$

it becomes tractable. This is called the *relaxed problem* (RP).

The key observation for RP is that the Lagrangian version

$$f^{\pi} + \nu g^{\pi} = \sum_{i=1}^{K} \left(\mathbf{E}\left[\sum_{t=0}^{\infty} c_i \mathbf{1}_{\{Z_i^{\pi}(t)>0\}}\right] + \nu\, \mathbf{E}\left[A^{\pi}(t)\right] \right),$$

where $\nu > 0$ is the Lagrange multiplier of the relaxed problem, is separable. This means that, for each user i, we can separately find a policy i that minimizes the objective function

$$f_i^{\pi_i} + \nu g_i^{\pi_i}, \tag{4.23}$$

where

$$f_i^{\pi_i} = \mathbf{E}\left[\sum_{t=0}^{\infty} c_i \mathbf{1}_{\{Z_i^{\pi_i}(t)>0\}}\right], \quad g_i^{\pi_i} = \mathbf{E}\left[\sum_{t=0}^{\infty} A^{\pi_i}(t)\right].$$

These subproblems can then be considered in the context of MDP as described in Publication IX.

Additionally, we define the indexability property as follows: The optimization problem (4.23) is said to be indexable if for any $j \in \{1, \ldots, J\}$ and $r \in \{r_{i,1}, r_{i,2}\}$, there exists $\nu^*(j,r) \in [0, \infty]$ such that

(i) it is optimal to schedule job i in state (j,r) if $\nu^*(j,r) \geq \nu$;

(ii) it is optimal not to schedule job i in state (j,r) if $\nu^*(j,r) \leq \nu$.

The quantity $\nu^*(j,r)$is called the Whittle index. Following this definition, in Theorem 1 of Publication IX, we prove the following:

The optimization problem (4.23) *is indexable, and the Whittle indices are given by*

$$\nu^*(j, r_{i,1}) = \frac{c_i}{q_i\left(\frac{r_{i,2}}{r_{i,2}} - 1\right)}, \qquad \nu^*(j, r_{i,2}) = \infty. \tag{4.24}$$

Thus, we observe that it is always optimal to serve a user if it is in its best possible channel state as its Whittle index is infinite in that state. However, this rule cannot be directly extended to the original scheduling

problem where only one user can be scheduled at a time. Therefore, we need a tie-breaking rule when there are multiple users in their best possible channel states so that only one user may be selected. To find such a tie-breaking rule, we consider the discounted version of RP, with the objective function

$$f_{i,\beta}^{\pi_i} + \nu g_{i,\beta}^{\pi_i}, \tag{4.25}$$

where $\beta \in (0,1)$, is the discount factor and

$$f_{i,\beta}^{\pi_i} = \mathbf{E}\left[\sum_{t=0}^{\infty} \beta^t c_i \mathbf{1}_{\{Z_i^{\pi_i}(t)>0\}}\right], \quad g_{i,\beta}^{\pi_i} = \mathbf{E}\left[\sum_{t=0}^{\infty} \beta^t A^{\pi_i}(t)\right].$$

This discounted problem can also be considered in the context of Markov decision processes as discussed in Publication IX. Furthermore, analogously as in the undiscounted case, we can define the indexability of this problem using Whittle index $\nu_{i,\beta}^* \in [0,\infty]$. We again show that the problem is indexable in Theorem 2 of Publication IX:

The optimization problem with objective function (4.25) *is indexable. In addition, for all* $j \in \{1,\ldots,J\}$*,*

$$\nu_{i,\beta}^*(j, r_{i,2}) \geq \nu_{i,\beta}^*(j, r_{i,1}),$$

and, for all $j \in \{2,\ldots,J\}$ *and* $r \in \{r_{i,1}, r_{i,2}\}$*,*

$$\nu_{i,\beta}^*(j-1, r) \geq \nu_{i,\beta}^*(j, r).$$

The Whittle indices are then determined according to Theorem 3 of Publication IX:

There is $\tilde{\beta} < 1$ *such that, for any* $\beta \in (\tilde{\beta}, 1)$*, the Whittle index for the optimization problem with the objective function* (4.23) *is given by*

$$\nu_{i,\beta}^*(j, r_1) = \frac{\frac{r_{i,1}}{\bar{r}_i}(\beta p_i \bar{r}_i)^j c_i}{(1-\beta+\beta p_i \bar{r}_i)^j - \frac{r_{i,1}}{\bar{r}_i}(\beta p_i \bar{r}_i)^j},$$

$$\nu_{i,\beta}^*(j, r_2) = \frac{\frac{1}{q_i}(\beta p_i q_i r_{i,2})^j c_i}{(1-\beta+\beta p_i r_{i,2})^j - (\beta p_i q_i r_{i,2})^j},$$

for any $j \in \{1,\ldots,J\}$.

Additionally, we also note that

$$\lim_{\beta\to 1} \nu_{i,\beta}^*(j, r_{i,1}) = \frac{c_i}{q_i\left(\frac{r_{i,2}}{r_{i,1}} - 1\right)} = \nu_i^*(j, r_{i,1}),$$

$$\lim_{\beta\to 1} \nu_{i,\beta}^*(j, r_{i,2}) = \infty = \nu_i^*(j, r_{i,2}),$$

$$\lim_{\beta\to 1} (1-\beta)\nu_{i,\beta}^*(j, r_{i,1}) = 0,$$

$$\lim_{\beta\to 1} (1-\beta)\nu_{i,\beta}^*(j, r_{i,2}) = \frac{c_i p_i r_{i,2}}{j}.$$

The last result is used to define the secondary index for the original job which has J phases at $t = 0$, as

$$\lim_{\beta \to 1} (1-\beta)\nu^*_{i,\beta}(J, r_{i,2}) = \frac{c_i r_{i,2}}{x_i},$$

This is used as the secondary index for making scheduling decisions for a job of size x_i in the heuristic policy we define next.

4.5.2 Heuristic for OP at the timeslot level

Inspired by the results for the relaxed problem when jobs are distributed according to a phase-type distribution, we define a heuristic size-aware Whittle index policy (SW) for OP as follows:
Assume that there are K jobs with remaining sizes y_i. For any $r \in \{r_{i,1}, r_{i,2}\}$, the primary index of job i is defined by

$$\nu_i^{\mathrm{SW}}(y_i, r_{i,1}) = \frac{c_i}{q_i \left(\frac{r_{i,2}}{r_{i,1}} - 1\right)}, \qquad \nu_i^{\mathrm{SW}}(y_i, r_{i,2}) = \infty, \tag{4.26}$$

and the secondary indices by

$$\tilde{\nu}_i^{\mathrm{SW}}(y_i, r_{i,1}) = 0, \qquad \tilde{\nu}_i^{\mathrm{SW}}(y_i, r_{i,2}) = \frac{c_i r_{i,2}}{y_i}. \tag{4.27}$$

The scheduling rule that

(i) gives an absolute priority to the jobs whose channel state is "good", breaking ties among these jobs by choosing the job i with the highest secondary index $\tilde{\nu}_i^{\mathrm{SW}}(y_i, r_{i,2})$, and

(ii) if there are no jobs whose channel state is "good", chooses the job i with the highest primary index $\nu_i^{\mathrm{SW}}(y_i, r_{i,1})$, breaking ties among these jobs randomly,

is called the size-aware Whittle index (SW) policy.

Note that for a homogeneous user population, SW always selects the user in its highest channel state, with the ties broken according to the SRPT principle. If none of the user is in its highest possible state, then one is chosen randomly for service.

4.5.3 Numerical results

In this section, we evaluate, by simulations, the performance of the proposed size-aware scheduler SW in a dynamic system with randomly varying number of jobs and i.i.d. channel variations. We demonstrate that SW

gives systematically better performance than other opportunistic scheduling policies like PF and PI defined below. Each of these policies can be defined by their own index rules, analogously as SW.

Consider a job i with a *random* size of X_i with mean $E[X_i] = 1/\mu_i$. Let a_i denote the attained service of job i (in bits) and d_i the time that it has already spent in the system (at the decision time). Thus, the throughput of job i at that time is equal to a_i/d_i. The PF rule uses only a single (primary) index, which depends on the current channel state and the throughput as

$$\nu_i^{\mathrm{PF}}(a_i, d_i, r_{i,1}) = \frac{r_{i,1}}{a_i/d_i}, \quad \nu_i^{\mathrm{PF}}(a_i, d_i, r_{i,2}) = \frac{r_{i,2}}{a_i/d_i}.$$

Note that this is the timeslot-level version of the flow-level PF operating point defined in Section 4.4.4. The PI policy [12] uses both the primary and the secondary indices. The primary index of this job under the PI policy is given by

$$\nu_i^{\mathrm{PI}}(r_{i,1}) = \frac{c_i}{q_i(\frac{r_{i,2}}{r_{i,1}} - 1)}, \quad \nu_i^{\mathrm{PI}}(r_{i,2}) = \infty,$$

and the secondary index by

$$\tilde{\nu}_i^{\mathrm{PI}}(r_{i,1}) = 0, \quad \tilde{\nu}_i^{\mathrm{PI}}(r_{i,2}) = c_i \mu_i r_{i,2}.$$

In what follows, we construct various scenarios for the study of SW and compare its performance against other scheduling rules discussed above. To validate our implementations of the other policies, we include two experiments, which coincide with earlier work ([12, Scenario 1] and [90, Scenario 2]). For each load value, the simulations are conducted for a duration that corresponds to approximately 1 000 000 job arrivals. Unless otherwise stated, the holding costs are assumed to accrue with rate $c = 1$ and the timeslot duration is 1 ms.

Two-state channels

Our following three scenarios are related to heterogeneous users with two different job classes $k = 1, 2$. In this case, subscript k refers to the job class. Here we assume that the job sizes are exponential.

For each class k, jobs are i.i.d. and the job size distribution is exponential with mean $E[X_k]$. The two-state channel model for class $k = 1$ is specified by parameters

$$r_{k,1} = 8.4\,\mathrm{kb}, \quad r_{k,2} = 33.6\,\mathrm{kb}, \quad q_k = 0.5. \tag{4.28}$$

These rates have been taken from [90, Table 1]. Moreover, new class-k jobs arrive according to a Poisson process with rate λ_k (arrivals per timeslot),

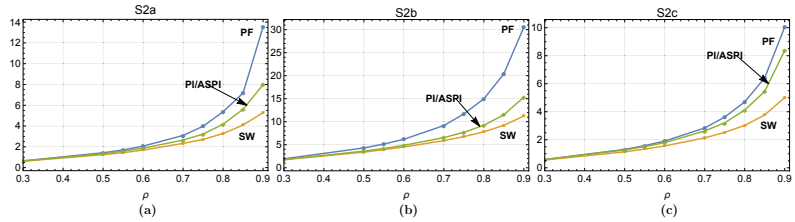


Figure 4.5. Mean holding cost incurred by both classes at different system load values, ρ, for scenarios S2a, S2b and S2c with heterogeneous users. Note that the Attained Service dependent Potential Improvement (ASPI) policy [90] is the same as PI in this case.

and the system load is denoted by $\rho = \rho_1 + \rho_2$, where

$$\rho_k = \lambda_k E[X_k]/r_{k,2}.$$

When the load is varying, we choose the arrival rates in the two classes so that the classwise loads remain equal ($\rho_1 = \rho_2$).

In our first scenario (S2a), we study the effect of the mean job size by having $E[X_1] = 0.5$ Mb and $E[X_2] = 5.0$ Mb. The channel model for class $k = 2$ is the same (4.28) as for class 1, as well as the holding cost rate ($c_1 = c_2 = 1$).

In our second scenario (S2b), we study the effect of different holding cost rates by having $c_1 = 5$ and $c_2 = 1$. The channel model for class $k = 2$ is again the same (4.28) as for class 1, as well as the mean job size ($E[X_1] = E[X_2] = 5.0$ Mb).

In our third scenario (S2c), we study the effect of different channel models by letting

$$r_{2,1} = 8.4\,\text{kb}, \quad r_{2,2} = 16.8\,\text{kb}, \quad q_2 = 0.5, \tag{4.29}$$

while the mean job sizes ($E[X_1] = E[X_2] = 5.0$ Mb) and the holding cost rates ($c_1 = c_2 = 1$) are the same in both classes. Again, the rates have been taken from [90, Table 1].

The results are given in Figure 4.5, which shows the mean holding costs as a function of the system load ρ for the three schedulers. We observe that SW is consistently better than the other two policies.

Multiple channel states

Our last scenario (S3) is the same as Scenario 1 in [12]. We consider heterogeneous users with two different job classes $k = 1, 2$, but now the channel state in each class has more than two possible states. The job sizes are geometric.

Let $q_{k,n} = P\{R_k(t) = r_{k,n}\}$, where $n \in \{1, \ldots, N_k\}$ and

$$r_{k,1} < \ldots < r_{k,N_k}.$$

We assume that $N_1 = 5$, $N_2 = 3$, and $r_{k,n}, q_{k,n}$ are taking values as given in [12, Table 1]. For each class k, jobs are i.i.d. and the job size distribution is geometric with mean $E[X_k] = 1/\mu_k = 102.57$ kb. New class-k jobs arrive according to a Poisson process with rate λ_k (arrivals per timeslot), and the system load is denoted by $\rho = \rho_1 + \rho_2$, where

$$\rho_k = \lambda_k E[X_k]/r_{k,N_k}.$$

The load of class 2 is kept constant $\rho_2 = 0.5$, while the load of class 1 is changed. In this experiment, the length of the timeslot is 1.67 ms.

Since there are now multiple possible channel states, we need to enlarge the definitions of the scheduling policies. The PF rule uses still a single (primary) index defined, for any $k \in \{1, 2\}$ and $n \in \{1, \ldots, N_k\}$, as follows:

$$\nu_k^{\mathrm{PF}}(a_i, d_i, r_{k,n}) = \frac{r_{k,n}}{a_i/d_i},$$

where a_i and d_i are defined as earlier for a job i belonging to class k. The primary index of the PI policy [12] is given, for each class k, by

$$\nu_k^{\mathrm{PI}}(r_{k,n}) = \frac{c_k}{\sum_{m>n} q_{k,m}(\frac{r_{k,m}}{r_{k,n}} - 1)}, \quad \nu_k^{\mathrm{PI}}(r_{k,N_k}) = \infty,$$

and the secondary index by

$$\tilde{\nu}_k^{\mathrm{PI}}(r_{k,n}) = 0, \quad \tilde{\nu}_k^{\mathrm{PI}}(r_{k,N_k}) = c_k \mu_k r_{k,N_k},$$

where $n \in \{1, \ldots, N_k - 1\}$. A natural extension of SW is defined by letting the primary index take values

$$\nu_k^{\mathrm{SW}}(y_i, r_{k,n}) = \frac{c_k}{\sum_{m>n} q_{k,m}(\frac{r_{k,m}}{r_{k,n}} - 1)}, \; \nu_k^{\mathrm{SW}}(y_i, r_{k,N_k}) = \infty,$$

and the secondary index values

$$\tilde{\nu}_k^{\mathrm{SW}}(y_i, r_{k,n}) = 0, \quad \tilde{\nu}_k^{\mathrm{SW}}(y_i, r_{k,N_k}) = \frac{c_k r_{k,N_k}}{y_i},$$

where $n \in \{1, \ldots, N_k - 1\}$ and y_i is the remaining size of job i belonging to class k (as earlier).

The results are given in Figure 4.6. Figure 4.6(a) shows the mean holding costs (i.e., the mean number of jobs in this case) as a function of the system load ρ for the three schedulers. Again, we observe that SW is performing better than both PF and PI for any load ρ. The gain comes

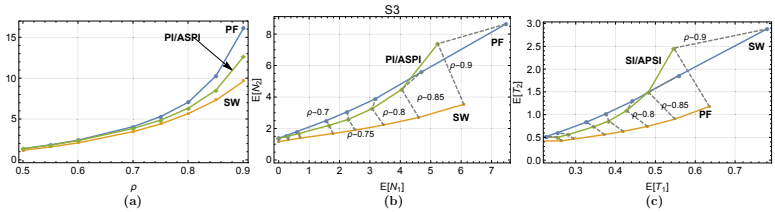


Figure 4.6. Performance of scenario S3 with multiple channel states. Panel (a) shows the mean holding cost or equivalently, in this case, the mean number of jobs at various values of system load, ρ. Panel (b) shows the plot of mean number of jobs in class 2 as a function of the mean number of jobs in class 1. Panel (c) shows the mean delay of class-2 jobs as a function of the mean delay of class-1 jobs. In panels (b) and (c), the isolines in gray color are drawn to indicate the same value of system load ρ. Note that the Attained Service dependent Potential Improvement (ASPI) policy [90] is the same as PI in this case.

mainly from the class that has fewer possible service rates (class 2 in this scenario). However, in the class with the larger number of possible service rates (class 1 in this scenario), the performances of these three schedulers are very close to each other except at very high loads, as seen from Figure 4.6(b).

In Figures 4.6(b) and 4.6(c), we show the indifference maps of the mean number of jobs and the mean delay of jobs, respectively, in the two classes of this scenario. The points in the dotted gray line are the values of mean number of jobs or delays corresponding to the same values of the load ρ. From Figure 4.6(b), we see that both PF and PI keep the number of jobs in both the classes fairly balanced, while the differences in the mean delays in the two classes are clearly the smallest for SW, as seen in Figure 4.6(c).

Finally we note that the results for PI in Figures 4.6 coincide with those in [12, Figures 1&3]. PF has not been evaluated in [12].

4.6 Summary

In this chapter, we considered the problem of scheduling the resources for data transmission to the users in a wireless system by taking into account their channel conditions, which can lead to opportunistic gain. However, maximizing such gain alone, when we also have the size information of the flows, will prevent the scheduler from taking advantage of gain from any SRPT-like policy. We have devised a policy that can provide the optimal trade-off between opportunistic gain and the gain obtained by prioritizing the user with smaller flow sizes, in a transient homogeneous system. This trade-off can be made at the flow level, assuming time

scale separation between flow-level dynamics and timeslot-level channel dynamics. Although, these optimality results apply for a transient system, they can be applied as heuristics to the system with dynamic arrivals, as well. However, we observe that the optimality property of such a policy is sometimes lost in dynamic systems, especially if the the opportunistic gain does not saturate quickly. Moreover, a simple policy that serves the users in their best possible channel state and breaks ties based on the SRPT principle appears to perform very well in a dynamic system.

When we have an asymmetric setting, where users have independent and heterogeneous two-state channels, the aforementioned optimality results do not hold anymore. In this case, by approximating the job sizes by a phase-type shifted Pascal distribution, we are able to define the problem as an RMAB problem and solve it using the Whittle index approach, which requires the scheduling constraints to be relaxed.

This approach, called the SW policy, is then applied as a heuristic to our original scheduling problem. SW schedules the user in its best possible channel state, and breaks the tie, if there are more than one users with such a channel state, using a secondary index. The secondary index, which applies the SRPT principle, is obtained by solving the discounted version of the relaxed scheduling problem. If none of the users is in its best possible state, one user is chosen randomly by SW. We observe that SW outperforms all other reference policies. Interestingly, with homogeneous users, SW is identical to SRPT-P when at least one user in its best channel. It outperforms even the TR-OPT policy in the dynamic system. The computation involved in applying this policy is very simple, and thus, if implemented, is able to provide gains even in the existing systems, if the system has perfect knowledge of the channel states.

5. Conclusions

In this thesis, we have studied three different aspects of the operation of the Radio Access Network (RAN) in a modern wireless system. Without going into very specific details about the operation of the network, we deal mainly with the abstract mathematical models of the network, making some simplifying assumptions that preserves the essence of the studied problems. Based on these studies, we have contributed some very general results, and even provided useful modeling techniques that are applicable to these networks. The results and some techniques used here can be applied, perhaps in the design of future networks, to provide better services to the user and to efficiently utilize the resources available to the network.

We started by studying the problem of congestion caused by the potentially massive amount of M2M traffic in LTE systems. Although LTE has been designed to handle large volumes of data traffic, we see that the signaling channel, PDCCH, becomes a bottleneck, and the random access process quickly becomes overwhelmed. Clearly, the current network design is unable to handle the deluge of service requests from billions of devices that are predicted to exist in future. Thus, future networks would have to take this fact into account for designing new random access protocols. For these networks, less complex implementation of the random access procedure or some form of hierarchical access scheme (e.g., the capillary network) could be a suitable option.

However, the congestion problem in LTE can be alleviated by using HetNets, which adds more capacity to the system. The low-powered femtos are able to serve the local traffic hotspots which could either be LTE or even WLANs. Thus, it is possible to reduce the access congestion by using various load balancing schemes, where the macro can take the excess traffic from the congested femto. We have seen that using dynamic policies,

which take the level of congestion in the cells into account, we can reduce the access delay. However, in this thesis, we have assumed that the level of congestion is always known to the load balancer. This may not always be a realistic assumption. On the other hand, with the development of software-based CRAN, which can maintain some form of central control over the network, it may be possible to estimate these quantities. The future research can be directed to these issues.

After the random access stage, the different base stations can coordinate their activities to transfer the actual data to users so that some gain is achieved by serving the right user by the right station. To study such coordination schemes, e.g., in dynamic TDD systems, we have utilized the flow-level models. By constructing policies based on different heuristics, we observe that coordinating the activities of neighboring bases stations does aid in reducing the flow level delays, especially if such policies take the number of users in the system into account. Again, we assume that the stations can share information for deciding the best policy. This form of information exchange should be possible between the stations with the advent of CRANs. Although it may be theoretically intractable, future work could consider such coordination between more than two cells through simulations.

When considering the downlink scheduling by just one base station to its users, we have seen that taking into account the size information of the user's flow to be sent, in addition to the channel state information, can be very advantageous. In such opportunistic scheduling problems, we have seen that relatively simple rules, with size information, are able to provide good gain over the non-anticipating policies. There are still much unanswered questions of both theoretical as well as practical nature. Although we have provided the optimal size-based policy in systems with homogeneous users and no user arrivals, we observe that these policies are not optimal when new users arrive. Characterizing the optimal policy in systems with dynamic arrivals is indeed a very hard problem, and the policy could be very sensitive to the channel model, which affects the geometry of the rate region.

We have also observed that the relaxation technique used to tackle the RMAB problem can also be applied to devise scheduling policies when the user population has heterogeneous channels. Although these optimal policies are developed for the system with the relaxed constraints and no new arrivals, they tend to provide good performance even in systems where

there are new arrivals. Surprisingly, with homogeneous users that have multi-state channels, the Whittle index-based heuristic can provide even better performance than the policy adapted from the optimality results with no user arrivals. We note that these policies first prioritize the users in their better channel states, and then use the size information only to break the ties. It will be interesting to observe if similar rule applies in general even when we have a more complicated channel model in future research.

References

[1] More than 50 billion connected devices. Technical report, Ericsson, 2011.

[2] Cisco visual networking index: Global mobile data traffic forecast update, 2014–2019. Technical report, Cisco, 2015.

[3] 3GPP. Study on RAN improvements for machine-type communications. TR 37.868, 3rd Generation Partnership Project (3GPP), 2011. V1.0.0 (2011-08).

[4] 3GPP. Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification. TS 36.321, 3rd Generation Partnership Project (3GPP), 2012. V11.0.0 (2012-09).

[5] S. Aalto and P. Lassila. Impact of size-based scheduling on flow-level performance in wireless downlink data channels. In *ITC 20*, pages 1096–1107, 2007.

[6] S. Aalto and P. Lassila. Flow-level stability and performance of channel-aware priority-based schedulers. In *NGI*, pages 1–8, 2010.

[7] N. Abramson. The Aloha system: another alternative for computer communications. In *AFIPS '70 (Fall)*, pages 281–285, 1970.

[8] A. M. Ahmadian, O. Galinina, S. Andreev, and Y. Koucheryavy. Modeling contention-based M2M transmissions over 3GPP LTE cellular networks. In *IEEE ICC Workshops*, pages 441–447, 2014.

[9] M. Al-Rawi and R. Jäntti. A dynamic TDD inter-cell interference coordination scheme for long term evolution networks. In *IEEE PIMRC*, pages 1590–1594, 2011.

[10] S. Andreev, M. Gerasimenko, O. Galinina, Y. Koucheryavy, N. Himayat, S.-P. Yeh, and S. Talwar. Intelligent access network selection in converged multi-radio heterogeneous networks. *IEEE Wireless Communications*, 21(6):86–96, 2014.

[11] S. Andreev, A. Larmo, M. Gerasimenko, V. Petrov, O. Galinina, T. Tirronen, J. Torsner, and Y. Koucheryavy. Efficient small data access for machine-type communications in LTE. In *IEEE ICC*, pages 3569–3574, 2013.

[12] U. Ayesta, M. Erausquin, and P. Jacko. A modeling framework for optimizing the flow-level scheduling with time-varying channels. *Performance Evaluation*, 67(11):1014–1029, 2010.

[13] U. Ayesta, M. Erausquin, M. Jonckheere, and I. M. Verloop. Scheduling in a random environment: stability and asymptotic optimality. *IEEE/ACM Transactions on Networking*, 21(1):258–271, 2013.

[14] P. Bender, P. Black, M. Grob, R. Padovani, N. Sindhushyana, and A. Viterbi. Cdma/hdr: a bandwidth efficient high speed wireless data service for nomadic users. *IEEE Communications Magazine*, 38(7):70–77, 2000.

[15] F. Berggren and R. Jäntti. Asymptotically fair transmission scheduling over fading channels. *IEEE Transactions on Wireless Communications*, 3(1):326–336, 2004.

[16] D. Bertsekas and R. Gallager. *Data Networks*. Prentice-Hall, second edition, 1992.

[17] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, 2000.

[18] T. Bonald. A score-based opportunistic scheduler for fading radio channels. In *European Wireless*, pages 1–7, 2004.

[19] T. Bonald, S. Borst, N. Hegde, M. Jonckheere, and A. Proutiere. Flow-level performance and capacity of wireless networks with user mobility. *Queueing Systems*, 63(1-4):131–164, 2009.

[20] T. Bonald and M. Feuillet. On the stability of flow-aware CSMA. *Performance Evaluation*, 67(11):1219–1229, 2010.

[21] T. Bonald and N. Hegde. Capacity gains of some frequency reuse schemes in OFDMA networks. In *IEEE GLOBECOM*, pages 4403–4408, 2009.

[22] T. Bonald, A. Ibrahim, and J. Roberts. The impact of association on the capacity of WLANs. In *WiOPT*, pages 1–10, 2009.

[23] T. Bonald and A. Proutière. Insensitivity in processor-sharing networks. *Performance Evaluation*, 49:193–209, 2002.

[24] T. Bonald and A. Proutière. Insensitive bandwidth sharing in data networks. *Queueing Systems*, 44:69–100, 2003.

[25] T. Bonald and A. Proutière. Wireless downlink data channels: user performance and cell dimensioning. In *ACM MobiCom*, pages 339–352, 2003.

[26] T. Bonald and A. Proutière. On performance bounds for balanced fairness. *Performance Evaluation*, 55:25–50, 2004.

[27] T. Bonald, A. Proutière, J. Roberts, and J. Virtamo. Computational aspects of balanced fairness. In *ITC 18*, pages 801–810, 2003.

[28] S. Borst. User-level performance of channel-aware scheduling algorithms in wireless data networks. *IEEE/ACM Transactions on Networking*, 13(3):636–647, 2005.

[29] S. Borst, N. Hegde, and A. Proutière. Interacting queues with server selection and coordinated scheduling — application to cellular data networks. *Annals of Operations Research*, 170:59–78, 2009.

[30] S. Borst and M. Jonckheere. Flow-level stability of channel-aware scheduling algorithms. In *WiOpt*, pages 1–6, 2006.

[31] F. Cecchi and P. Jacko. Scheduling of users with Markovian time-varying transmission rates. *ACM SIGMETRICS Performance Evaluation Review*, 41(1):129–140, 2013.

[32] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann. Cloud RAN for mobile networks—a technology overview. *IEEE Communications Surveys & Tutorials*, 17(1):405–426, 2014.

[33] K.-C. Chen and S.-Y. Lien. Machine-to-machine communications: Technologies and challenges. *Ad Hoc Networks*, 18:3–23, 2014.

[34] M. Chen, J. Wan, S. Gonzalez, X. Liao, and V. Leung. A survey of recent developments in home M2M networks. *IEEE Communications Surveys Tutorials*, 16(1):98–114, 2014.

[35] J.-P. Cheng, C.-h. Lee, and T.-M. Lin. Prioritized Random Access with dynamic access barring for RAN overload in 3GPP LTE-A networks. In *IEEE GLOBECOM Workshops*, pages 368–372, 2011.

[36] C. Chiang, W. Liao, T. Liu, I. Chan, and H. Chao. Adaptive downlink and uplink channel split ratio determination for TCP-based best effort traffic in TDD-based WiMAX networks. *IEEE Journal Selected Areas in Communications*, 27(2):182–190, 2009.

[37] S. Choi, W. Lee, D. Kim, K.-J. Park, S. Choi, and K.-Y. Han. Automatic configuration of random access channel parameters in LTE systems. In *WD*, pages 1–6, 2011.

[38] J. Conti. The internet of things. *Communications Engineer*, 4(6):20–25, 2006.

[39] E. Dahlman, G. Mildh, S. Parkvall, J. Peisa, J. Sachs, and Y. Selén. 5G radio access. *Ericsson Review*, 6:2–7, 2014.

[40] E. Dahlman, S. Parkvall, and J. Sköld. *4G LTE/LTE-Advanced for Mobile Broadband*. Academic Press, 2011.

[41] A. Damnjanovic, J. Montojo, Y. Wei, T. Ji, T. Luo, M. Vajapeyam, T. Yoo, O. Song, and D. Malladi. A survey on 3GPP heterogeneous networks. *IEEE Wireless Communications*, 18(3):10–21, 2011.

[42] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao. 5G on the horizon: key challenges for the radio-access network. *IEEE Vehicular Technology Magazine*, 8(3):47–53, 2013.

[43] D. C. Dimitrova, H. van den Berg, G. Heijenk, and R. Litjens. Flow level performance comparison of packet scheduling schemes for UMTS EUL. In *WWIC*, pages 27–40, 2008.

[44] G. Fayolle, E. Gelenbe, and J. Labetoulle. Stability and optimal control of the packet switching broadcast channel. *Journal of the ACM*, 24(3):375–386, 1977.

[45] O. Galinina, A. Turlikov, S. Andreev, and Y. Koucheryavy. Stabilizing multi-channel slotted aloha for machine-type communications. In *IEEE ISIT*, pages 2119–2123, 2013.

[46] J. Gozalvez. Tentative 3GPP timeline for 5G [mobile radio]. *IEEE Vehicular Technology Magazine*, 10(3):12–18, 2015.

[47] B. Hajek. Optimal control of two interacting service stations. *IEEE Transactions on Automatic Control*, 29(6):491–499, 1984.

[48] E. Hyytiä, J. Virtamo, S. Aalto, and A. Penttinen. M/M/1-PS queue and size-aware task assignment. *Performance Evaluation*, 68(11):1136–1148, 2011.

[49] P. Jacko. Value of information in optimal flow-level scheduling of users with Markovian time-varying channels. *Performance Evaluation*, 68(11):1022–1036, 2011.

[50] A. Jalali, R. Padovani, and R. Pankaj. Data throughput of cdma-hdr a high efficiency-high data rate personal communication wireless system. In *IEEE VTC 2000-Spring*, volume 3, pages 1854–1858, 2000.

[51] F. Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37, 1997.

[52] J. Kim, B. Kim, J. Kim, and Y. H. Bae. Stability of flow-level scheduling with Markovian time-varying channels. *Performance Evaluation*, 70(2):148–159, 2013.

[53] L. Kleinrock and S. Lam. Packet switching in a multiaccess broadcast channel: Performance evaluation. *IEEE Transactions on Communications*, 23(4):410–423, 1975.

[54] A. Larmo and R. Susitaival. RAN overload control for Machine Type Communications in LTE. In *IEEE GLOBECOM Workshops*, pages 1626–1631, 2012.

[55] P. Lassila and S. Aalto. Combining opportunistic and size-based scheduling in wireless systems. In *ACM MSWiM*, pages 323–332, 2008.

[56] P. Lassila, A. Penttinen, and S. Aalto. Flow-level modeling and analysis of dynamic TDD in LTE. In *NGI*, pages 33–40, 2012.

[57] K.-D. Lee, S. Kim, and B. Yi. Throughput comparison of random access methods for M2M service over LTE networks. In *IEEE GLOBECOM Workshops*, pages 373–377, 2011.

[58] J. Leino and J. Virtamo. Determining the moments of queue-length distribution of discriminatory processor-sharing systems with phase-type service requirements. In *NGI*, pages 205–208, 2007.

[59] S.-Y. Lien, K.-C. Chen, and Y. Lin. Toward ubiquitous massive accesses in 3GPP machine-to-machine communications. *IEEE Communications Magazine*, 49(4):66–74, 2011.

[60] J. Liu, A. Proutière, Y. Yi, M. Chiang, and H. V. Poor. Flow-level stability of data networks with non-convex and time-varying rate regions. *ACM SIGMETRICS Performance Evaluation Review*, 35(1):239–250, 2007.

[61] X. Liu, E. K. Chong, and N. B. Shroff. A framework for opportunistic scheduling in wireless networks. *Computer Networks*, 41(4):451–474, 2003.

[62] M. J. Marcus. 5G and IMT for 2020 and beyond. *IEEE Wireless Communications*, 22(4):2–3, 2015.

[63] J. Mo and J. Walrand. Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking*, 8(5):556–567, 2000.

[64] T. Nakamura, A. Benjebbour, Y. Kishiyama, and S. Suyama. 5G radio access: Requirements, concept and experimental trials. *IEICE Transactions on Communications*, 98(8):1397–1406, 2015.

[65] T. Nakamura, S. Nagata, A. Benjebbour, Y. Kishiyama, T. Hai, S. Xiaodong, Y. Ning, and L. Nan. Trends in small cell enhancements in LTE advanced. *IEEE Communications Magazine*, 51(2):98–105, 2013.

[66] A. Osseiran, F. Boccardi, V. Braun, K. Kusume, P. Marsch, M. Maternia, O. Queseth, M. Schellmann, H. Schotten, H. Taoka, et al. Scenarios for 5G mobile and wireless communications: the vision of the METIS project. *IEEE Communications Magazine*, 52(5):26–35, 2014.

[67] D. Park, H. Seo, H. Kwon, and B. G. Lee. Wireless packet scheduling based on the cumulative distribution function of user transmission rates. *IEEE Transactions on Communications*, 53(11):1919–1929, 2005.

[68] A. Penttinen, J. Virtamo, and R. Jäntti. Performance analysis in multi-hop radio networks with balanced fair resource sharing. *Telecommunication Systems*, 31:315–336, 2006.

[69] M. Pinedo. *Scheduling: Theory, Algorithms, and Systems*. Springer Science & Business Media, 2008.

[70] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[71] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez. Millimeter wave mobile communications for 5G cellular: It will work! *IEEE Access*, 1:335–349, 2013.

[72] S. Rayment and J. Bergström. Achieving carrier-grade Wi-Fi in the 3GPP world. *Ericsson Review*, pages 2–7, 2012.

[73] B. Rengarajan. *Self Organizing Networks: Building Traffic and Environment Aware Wireless Systems*. PhD thesis, The University of Texas at Austin, 2009.

[74] B. Rengarajan and G. de Veciana. Architecture and abstractions for environment and traffic-aware system-level coordination of wireless networks. *IEEE/ACM Transactions on Networking*, 19(3):721–734, 2011.

[75] M. Rivero-Angeles, D. Lara-Rodriguez, and F. Cruz-Perez. A new EDGE medium access control mechanism using adaptive traffic load slotted ALOHA. In *IEEE VTS VTC 2001 Fall*, volume 3, pages 1358–1362, 2001.

[76] L. G. Roberts. ALOHA packet system with and without slots and capture. *ACM SIGCOMM Computer Communication Review*, 5(2):28–42, 1975.

[77] W. Rosenkrantz and D. Towsley. On the instability of slotted ALOHA multiaccess algorithm. *IEEE Transactions on Automatic Control*, 28(10):994–996, 1983.

[78] P. Rost, C. Bernardos, A. Domenico, M. Girolamo, M. Lalam, A. Maeder, D. Sabella, et al. Cloud technologies for flexible 5G radio access networks. *IEEE Communications Magazine*, 52(5):68–76, 2014.

[79] B. Sadiq and G. de Veciana. Balancing SRPT prioritization vs opportunistic gain in wireless systems with flow dynamics. In *ITC 22*, pages 1–8, 2010.

[80] L. Schrage. Letter to the editor—a proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 16(3):687–690, 1968.

[81] J.-B. Seo and V. Leung. Design and analysis of backoff algorithms for random access channels in UMTS-LTE and IEEE 802.16 systems. *IEEE Transactions on Vehicular Technology*, 60(8):3975–3989, 2011.

[82] J.-B. Seo and V. Leung. Performance modeling and stability of semi-persistent scheduling with initial random access in LTE. *IEEE Transactions on Wireless Communications*, 11(12):4446–4456, 2012.

[83] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang. A first look at cellular machine-to-machine traffic: Large scale measurement and characterization. *ACM SIGMETRICS Performance Evaluation Review*, 40(1):65–76, 2012.

[84] X. Shen. Device-to-device communication in 5G cellular networks. *IEEE Network*, 29(2):2–3, 2015.

[85] Z. Shen, A. Khoryaev, E. Eriksson, and X. Pan. Dynamic uplink-downlink configuration and interference management in TD-LTE. *IEEE Communications Magazine*, 50(11):51–59, 2012.

[86] A. Stolyar. Maxweight scheduling in a generalized switch: state space collapse and workload minimization in heavy traffic. *Annals of Applied Probability*, 14(1):1–53, 2004.

[87] R. Susitaival, H. Wiemann, J. Östergaard, and A. Larmo. Internet access performance in LTE TDD. In *IEEE VTC 2010 Spring*, pages 1–5, 2010.

[88] W. Szpankowski. Packet switching in multiple radio channels: Analysis and stability of a random access system. *Computer Networks*, 7(1):17–26, 1983.

[89] I. Taboada, P. Jacko, U. Ayestaa, and F. Liberal. Opportunistic scheduling of flows with general size distribution in wireless time-varying channels. In *ITC 26*, pages 1–9, 2014.

[90] I. Taboada, F. Liberal, and P. Jacko. An opportunistic and non-anticipating size-aware scheduling proposal for mean holding cost minimization in time-varying channels. *Performance Evaluation*, 79:90–103, 2014.

[91] T. Taleb. Toward carrier cloud: Potential, challenges, and solutions. *IEEE Wireless Communications*, 21(3):80–91, 2014.

[92] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[93] B. S. Tsybakov. File transmission over wireless fast fading downlink. *IEEE Transactions on Information Theory*, 48(8):2323–2337, 2002.

[94] H. van den Berg, R. Litjens, and J. Laverman. HSDPA flow level performance: the impact of key system and traffic aspects. In *ACM MSWiM*, pages 283–292, 2004.

[95] I. Verloop and R. Núñez-Queija. Asymptotically optimal parallel resource assignment with interference. *Queueing Systems*, 65(1):43–92, 2010.

[96] C.-X. Wang, F. Haider, X. Gao, X.-H. You, Y. Yang, D. Yuan, H. Aggoune, H. Haas, S. Fletcher, and E. Hepsaydir. Cellular architecture and key technologies for 5G wireless communication networks. *IEEE Communications Magazine*, 52(2):122–130, 2014.

[97] G. Wang, X. Zhong, S. Mei, and J. Wang. An adaptive medium access control mechanism for cellular based Machine to Machine (M2M) communication. In *IEEE ICWITS*, pages 1–4, 2010.

[98] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, pages 287–298, 1988.

Errata

Publication IV

- The first sentence of the caption of Figure 2 should read:
  Plot of the optimal rate pair (r_1^*, r_2^*) against the distance parameter k for asymmetrical arrival rates $(\lambda_{11} = 10, \lambda_{12} = 1, \lambda_{21} = 19, \lambda_{22} = 1, c_0 = 50, c_1 = 30)$.

- In Section V, the maximum service rates are $c_0 = 50, c_1 = 30$, for all cases.

Publication V

- The last sentence of the caption of Figure 3 should read:
  The legend in panel (a) shows the dynamic policies involved in Scenario 2 and the legend in panel (c) shows the different dynamic policies involved in Scenario 3.

Publication VI

- Equation (1) is valid only when the modes change very slowly compared to the rate at which the time slots to the flows are allocated. In any dynamic operating mode, this time-scale separation assumption is valid,

and thus (1) is strictly applicable only in the dynamic modes.

- Equation (7) gives the service rates of the different queues in a static mode. In a static policy the operating modes are probabilistically chosen in each time slot. As the modes change very quickly compared to the flow-level dynamics, these flows receive an average service rate that depends on the probability distribution of the different modes. Since there are two possible modes that serve the downlink queues, the expressions presented in (7) are approximates of the actual downlink service rate in this case. The actual expression of the downlink service rates ϕ_i^{d} are as follows[1]

$$\frac{1}{\phi_1^{\mathrm{d}}} = \mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{p^{\mathrm{dd}}c_1^{\mathrm{dd}}(R_1) + p^{\mathrm{du}}c_1^{\mathrm{du}}(R_1)}\right],$$

$$\frac{1}{\phi_2^{\mathrm{d}}} = \mathbf{E}[B_i^{\delta}]\,\mathbf{E}\left[\frac{1}{p^{\mathrm{dd}}c_2^{\mathrm{dd}}(R_2) + p^{\mathrm{ud}}c_2^{\mathrm{ud}}(R_2)}\right].$$

  However, since the uplink queues are served by only one mode, the expressions for the uplink service rates ϕ_i^{u} are the same as presented in (7).

  When these actual service rates of the physical model are used, the resulting capacity region seems to be larger than when the approximate values are used. This can be observed from an example in Figure 5.1.

  The physical model optimal static service rate values can be obtained by conducting a local search around the approximated values. For the scenarios considered in the paper, the approximated values tend to underestimate the service rates of the queues, and therefore the average queue lengths are smaller when the physical model service rates are used. The approximation error tends to grow with increasing load values and appears to be very small when the fraction of the arrival rate that can be stabilized is small.

[1] cf. B. Rengarajan and G. de Veciana. Architecture and abstractions for environment and traffic-aware system-level coordination of wireless networks. *IEEE/ACM Transactions on Networking*, 19(3):721–734, 2011.

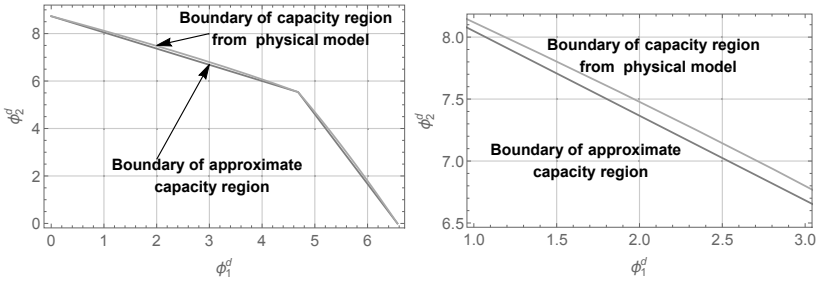


Figure 5.1. The capacity region from the physical model and the same obtained from (3.14) when $p^{\mathrm{uu}} = 0.5$ for Scenario 3 described in Section 8.5. Note that the difference is discernible only in the magnified picture (seen in the right panel). We can also see that the capacity region obtained from the physical model is clearly a superset of the one approximated by (7).

Publication IX

- In p. 60 the left side of the third equation from last should read:

$$c_1 + (1 - q_i)(\nu - p_i r_{i,1} \Delta_i(\nu)) + q_i(\nu - p_i r_{i,2} \Delta_i(\nu))$$

This thesis summarizes the works related to three different aspects of resource allocation at the radio access nework, which are useful in the design of fifth generation (5G) wireless networks.

First, the random access procedure of LTE is analyzed when massive number of machine-to-machine (M2M) users are present. The causes of congestion are investigated and possible remedies using heterogeneous networks (HetNets) is suggested.

The problem of intercell coordination between interfering base stations is then considered. In different problem settings, various policies to mitigate to this interference that aid in minimizing delays of the users are studied.

Finally, the problem of scheduling downlink users in a cell is analyzed when the station is fully aware of the channel conditions of the users and the size of their jobs. The aim again is to minimize the delay.

Aalto-DD 137/2016

ISBN 978-952-60-6915-9 (printed)
ISBN 978-952-60-6914-2 (pdf)
ISSN-L 1799-4934
ISSN 1799-4934 (printed)
ISSN 1799-4942 (pdf)

Aalto University
School of Electrical Engineering
Department of Communications and Networking
www.aalto.fi

BUSINESS + ECONOMY

ART + DESIGN + ARCHITECTURE

SCIENCE + TECHNOLOGY

CROSSOVER

DOCTORAL DISSERTATIONS